

AD-A169 006

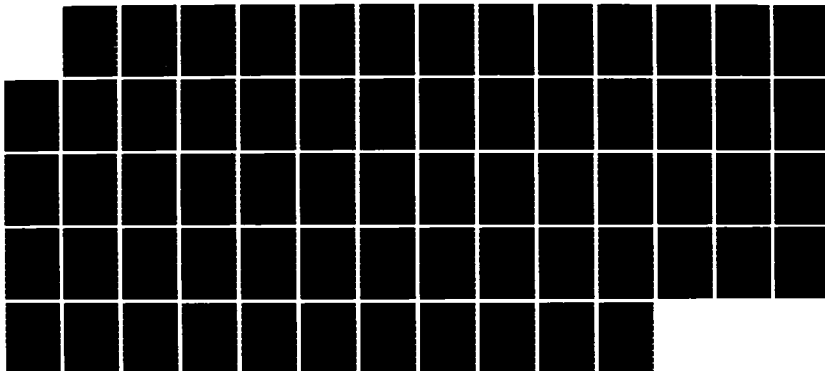
DEPARTMENT OF DEFENSE GATEWAY ARCHITECTURE AND
FUNCTIONAL REQUIREMENTS(U) SPARTA INC MCLEAN VA
11 DEC 85 URM-5-DCA-043 DCA100-04-C-0085

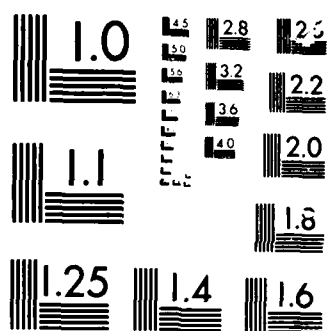
1/1

UNCLASSIFIED

F/G 17/2

NL





AD-A169 086

DTIC FILE COPY

2

DTIC
ELECTED
CONFIDENTIAL
FI



This document is not approved
for public release, its
distribution is unlimited.

Department of Defense Gateway Architecture
and Functional Requirements

Submitted by

SPARTA, INC.
7926 Jones Branch Drive, Suite 1070
McLean, VA 22102

Submitted To

Defense Communication Engineering Center
1860 Wiehle Ave.
Reston, VA

For Contract
DCA100-84-C-0085

DTIC
SECRET
E

December 22, 1985

This document has been approved
for public release and sale; its
distribution is unlimited.

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE

ADA 169086

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188
Exp. Date Jun 30 1995

1a REPORT SECURITY CLASSIFICATION UNCLASSIFIED			1b RESTRICTIVE MARKINGS N/A	
2a SECURITY CLASSIFICATION AUTHORITY			3 DISTRIBUTION/AVAILABILITY OF REPORT UNLIMITED	
2b DECLASSIFICATION/DOWNGRADING SCHEDULE				
4 PERFORMING ORGANIZATION REPORT NUMBER(S) URW-5-DCA-043			5 MONITORING ORGANIZATION REPORT NUMBER(S)	
6a NAME OF PERFORMING ORGANIZATION SPARTA, INC.		6b OFFICE SYMBOL (If applicable)	7a NAME OF MONITORING ORGANIZATION Defense Communications Engineering Center	
6c ADDRESS (City, State, and ZIP Code) 7926 Jones Branch Drive, Suite 1070 McLean, VA 22102			7b ADDRESS (City, State, and ZIP Code) 1860 Wiehle Ave Reston, VA 22090	
8a NAME OF FUNDING/SPONSORING ORGANIZATION DCEC		8b OFFICE SYMBOL (If applicable) R640	9 PROCUREMENT INSTRUMENT IDENTIFICATION NUMBER DCA100-84-C-0085	
8c ADDRESS (City, State, and ZIP Code) 1860 Wiehle Ave Reston, VA 22090			10 SOURCE OF FUNDING NUMBERS	
			PROGRAM ELEMENT NO	PROJECT NO
11 TITLE (Include Security Classification) Department of Defense Gateway Architecture and Functional Requirements				
12 PERSONAL AUTHOR(S) SPARTA, INC.				
13a TYPE OF REPORT FINAL		13b TIME COVERED FROM _____ TO _____		14 DATE OF REPORT (Year, Month, Day) 851211
15 PAGE COUNT 53				
16 SUPPLEMENTARY NOTATION				
17 COSATI CODES			18 SUBJECT TERMS (Continue on reverse if necessary and identify by block number) Computer Networks Defense Data Network Internetworking Gateway	
FIELD	GROUP	SUB GROUP		
19 ABSTRACT (Continue on reverse if necessary and identify by block) <p>This document presents the functional requirements for DoD Internetwork gateways. The requirements themselves are highlighted in Sections 3 and 5 through 7. In addition, Section 1 discusses the motivations for these requirements in terms of a Gateway's role the DoD Internetwork; Section 2 discusses successful gateway designs. Section 4 illuminates several gateway design tradeoff areas, where design options can be used to match specific operational and performance requirements.</p>				
20 DISTRIBUTION/AVAILABILITY OF ABSTRACT <input checked="" type="checkbox"/> UNCLASSIFIED/UNLIMITED <input type="checkbox"/> SAME AS RPT <input type="checkbox"/> DTIC USERS			21 ABSTRACT SECURITY CLASSIFICATION UNCLASSIFIED	
22a NAME OF RESPONSIBLE INDIVIDUAL Mr. Ed Cain			22b TELEPHONE (Include Area Code) (703) 437-2578	22c OFFICE SYMBOL R640

TABLE OF CONTENTS

1.0	GATEWAY ROLES IN THE INTERNET ARCHITECTURE	1
1.1	Introduction	1
1.2	Gateway Role In Internetworking	3
1.3	Internetworking Techniques	5
1.3.1	Generic Functions	5
1.3.2	Protocols And Techniques For Internetworking	6
1.3.3	Protocols And Techniques For Internetwork End To End Services	7
1.3.4	Internetwork Organization Concepts	8
1.4	Internet Architecture's Capabilities	10
1.5	Internet Architecture's Limitations	10
1.6	Internetworking In Non-DoD Environments	11
1.7	Gateway Mission In Internetwork Architecture	13
2.0	REVIEW OF EXISTING GATEWAY DESIGNS	14
2.1	The DARPA Internet Gateway	14
2.2	The MITRE Gateway	15
2.3	The Ford Aerospace Communication Corporation Multinet Gateway	16
2.4	The Bolt, Beranek And Newman Butterfly Gateway	18
2.5	EGP Under Berkeley 4.2 UNIX (DEC VAX)	21
2.6	SDC-Burroughs Gateway	22
2.7	The SAC Gateway	23
3.0	GATEWAY FUNCTIONAL REQUIREMENTS AND PROTOCOLS	24
3.1	Software Requirements	25
3.2	Hardware Functional Requirements	27
3.3	Basic Operational Requirements	28
4.0	GATEWAY DESIGN TRADEOFFS	28
4.1	Local And Remote Monitor/Control Capabilities	29
4.1.1	Monitor Capabilities Of Baseline Protocol Suite	30
4.1.2	Host Monitoring Protocol And XNET	30
4.1.3	FACC Multinet Gateway Local Monitoring	32
4.1.4	Capabilities Of Internetwork Operation Centers	32
4.1.5	Future Internet Management And Control Techniques	34
4.1.6	Tradeoff Discussion	35
4.2	Network Interfaces	36
4.3	Enhanced Services For Applications	37
4.4	Software Configuration Flexibility Versus Simplicity	38
4.5	Interior Gateway Protocols And Routing Algorithms	39
4.6	Gateway Architecture	40
4.7	Gateway Survivability	42
4.8	Gateway Physical Characteristics	43
4.9	Gateway Reliability And Maintainability	43
4.10	Gateway Operational Controls	44
5.0	GATEWAY MONITORING AND FAULT ISOLATION	46
5.1	Local Monitor And Control Capabilities	47
5.2	Remote Monitoring And Control Capabilities	48
6.0	INTERACTION AMONG SYSTEMS OF GATEWAYS	48
7.0	GATEWAY SECURITY CONSIDERATIONS	50

Accession For
NTIS GRA&I
DTIC TAB
Unannounced
Justification
By
Distribution /
Availability Codes
Avail and/or
Dist Special
A-1



Summary

This document presents the functional requirements for DoD Internetwork gateways. The requirements themselves are highlighted in Sections 3 and 5 through 7. In addition, Section 1 discusses the motivations for these requirements in terms of a Gateway's role in the DoD Internetwork; Section 2 discusses successful gateway designs. Section 4 illuminates several gateway design tradeoff areas, where design options can be used to match specific operational and performance requirements.

1.0 GATEWAY ROLES IN THE INTERNET ARCHITECTURE

1.1 Introduction

During the past two decades, the use of packet switching for data communications between computers has grown tremendously. There have been several stages of this growth that have contributed to the need for internetwork gateways. Networks are organizations of packet switching computers and transmission media that present a single well-defined data communication service to attached user machines, or "hosts." The first packet-switched networks provided coverages of wide geographical areas using point-to-point communication lines connected via packet-switching machines. By breaking longer data messages into smaller packets, simultaneous virtual continuous service could be provided to a large number of connected machines. More recently, a second generation of "local area" networks has developed using a wider variety of architectures and transmission media. Consequently the need for internetwork communications has developed. Machines on one network need to communicate with machines on another, despite dissimilar transmission media and protocols and dissimilar network access protocols.

The term "Internetworking" refers to the strategy for meeting this need to allow hosts on distinct networks to communicate. Such a strategy consists of the following:

1. the topology and components for the interconnection; in the case of the DoD/DARPA, the gateway is the major new addition to the hosts and networks;
2. the internet service--the external behavior of the data communication and related services provided across multiple networks; and
3. the internetworking methods--the protocols and algorithms used by the network components to effect data transmission over multiple networks.

The DoD/DARPA Internetworking techniques have been in development for the past decade and have recently made the transition from experimental to production use. Currently more than eighty (80) gateways are operational in the MILNET and ARPANET environments.

There are multiple reasons for DoD data communications needs being met by multiple networks. Among them are the following:

1. Networks tend to be organized in accordance with a single underlying communication technology. For example a network is more likely to be based solely upon medium speed terrestrial wire links than upon both terrestrial wire and high-speed cable links. Technical reasons account in large part for the development of networks along lines of link technologies. Satellite links have long propagation times with high bandwidth and so behave differently than contention-based and reservation-based short haul networks; terrestrial point-to-point multi-node networks and links behave differently than either of these, and so forth. It is easier to design protocols to handle a single behavior type than multiple behavior types.
2. Earlier networks were based upon terrestrial wire links, while there has been a recent proliferation of high-speed cable networks.
3. There are motivations for multiple administrative domains with regard to networks. A command or organization can take the lead in setting up a network, obtaining functions and performance best suited to its users; yet there is a DoD-wide need for interoperability among systems that can and do communicate. Internetworking satisfies both of these motivations.

Gateways and the techniques they use to provide internetwork services are the focus of this report. It reviews the techniques used in gateway implementations in order to illustrate the design principles. It discusses gateway software in terms of standard protocols for communications and in terms of resident operating system support. It discusses gateway hardware requirements in terms of the basis of existing machines that are acting as gateways and in terms of design of future gateway machines. Finally, it discusses tradeoffs in gateway designs as they relate to choices of non-minimal functions and non-minimal performance.

The remainder of this section describes the environment in which gateways operate, concluding with an overview of gateway roles and functions. Section 2 reviews existing gateway designs; section 3 reviews the functions and protocols necessary for gateway operation. Section 4 addresses design tradeoffs for gateways. Section 5 discusses gateway capabilities for the support of fault isolation and monitoring. Section 6 discusses requirements for interactions among gateways, principally to exchange global internetwork status information. Section 7 discusses gateway security techniques.

1.2 Gateway Role In Internetworking

The DoD/DARPA approach makes a gateway a network host on two or more networks. A gateway follows the procedures and protocols for sending data over each of its networks, and the networks need not distinguish a gateway from any other hosts attached to them. (The advantages of this approach are clear: no network administration is required to re-work its architecture or software in order to accomodate internetwork communications.)

Two end-user hosts who wish to communicate over multiple networks employ both the protocols of their immediate networks and specific end-to-end protocols. The sender, realizing that the destination is on a distant network, addresses data instead to the gateway. When the gateway receives the data, it determines that the data must be sent across another of its networks, either to the ultimate destination, or perhaps to yet another gateway, and it effects that transmission. Therefore, the gateway's main functions are relaying and routing.

In effect, the gateway represents a packet switch, while the networks represent communication links. However, the DoD internetworking approach demands that gateways provide very limited packet switching services. There is very little error recovery and no tracking of packets for possible retransmissions. Gateways have relatively few routing decision powers. To provide adequate performance, the end users are expected to provide end-to-end error recovery by maintaining careful tracking of the packets (called "datagrams") exchanged via gateways over multiple networks.

Figure 1-1 illustrates the role of a gateway in an internetwork environment.

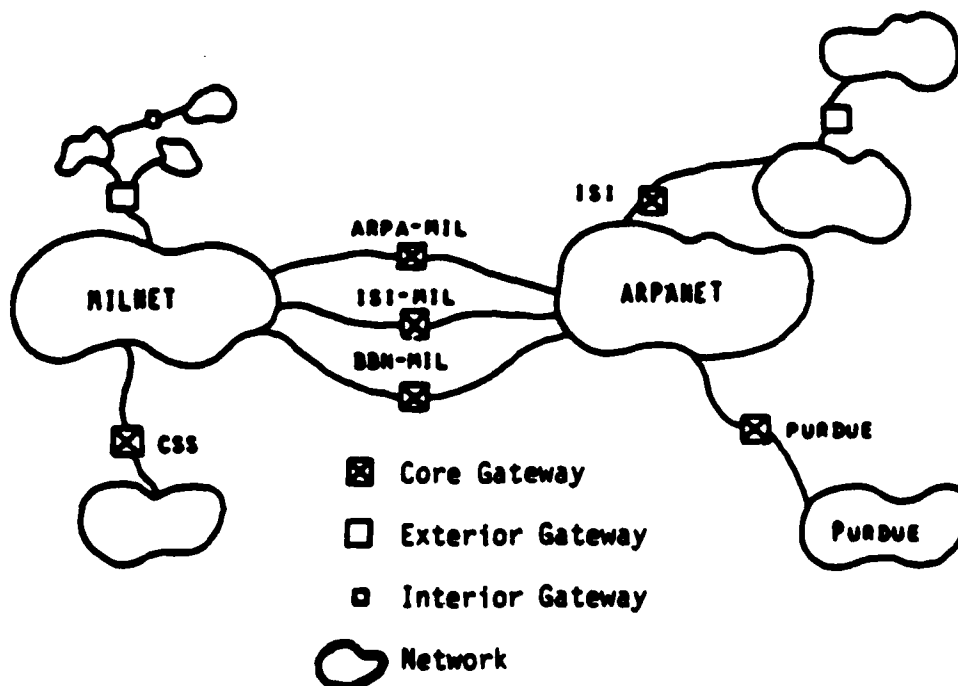


Figure 1-1. Gateways in the DoD Internetwork Topology

Physically the gateway is a computer that is capable of acting as a network host on its attached networks. (It is also possible to implement gateway functions through cooperation of two separate machines that are hosts on separate networks. These machines would be connected through a dedicated, non-network communication link. Such a link would either be of low bandwidth or else very expensive. The focus here is on single-chassis gateways.) In principle a gateway implementation may be based upon processors as simple as the Zilog Z80 to ones as powerful as a Cray XMP (TM). (XMP is a trademark of Cray Research, Inc.) Its switching throughput as an element of a communication path will be determined largely by how fast the processor can execute instructions

and by how fast the processor can perform memory accesses and input/output operations. In contrast, the size of primary and secondary storage is not as important, because both the size of the stored programs for packet switching and the size of the data units are relatively small (as compared to large scientific application programs that operate on huge arrays).

Historically, gateways have most often been implemented in Digital Equipment Corporation LSI-11 machines. These exemplify the lower end of processor speed, memory bandwidth and memory size. More recently, the Motorola 68000 has been used for gateway implementations, providing increased processor speed and memory bandwidth. (The 68000 can also address a larger memory space.) A typical 68000-based gateway may occupy a box capable of residing on a desk top, drawing on the order of 100 watts of power. If it is to serve only dedicated gateway functions, no secondary disk is necessary for storage of executable code; instead, the code can be stored in read-only memory (ROM). In other words, a gateway can be a compact, stand-alone microcomputer-based configuration with complexity and cost similar to a personal computer.

1.3 Internetworking Techniques

1.3.1 Generic Functions

Gateways and hosts must cooperate to provide a set of services that together provide a useable internetwork communication capability. These services may be summarized as follows:

1. Addressing and Routing: an internetwork-wide addressing scheme is required for "labeling" and finding internetwork participants; data traveling through the internetwork must make its way from source to destination; methods to accomplish this may vary; in many cases the route may be based upon the address of the destination;
2. Fragmentation and Reassembly: data packet sizes can vary from one separate network to another; data traveling from a large maximum packet size network into a small packet size network needs to be fragmented and properly reassembled at its destination;
3. Flow and Congestion Control: Hosts and networks protect themselves through flow and congestion control techniques; additional techniques are necessary to protect internetwork components, particularly gateways from these effects;
4. Error Checking and Reporting: Hosts and Networks typically use error checking and reporting techniques to support reliable communications over single networks; additional techniques are needed for additional internetworking components;

5. Internetwork Management: the internetwork requires monitoring and management, as does a single network; methods are needed for reporting status and statistics, exchanging information between gateways and monitoring centers, etc.

1.3.2 Protocols And Techniques For Internetworking

Largely under DARPA sponsorship, a family of protocols for interworking has developed to a mature state. The underlying architecture of this protocol family treats each network as a sovereign entity and as an intermediate carrier of data traffic. In fact, there is a clear analogy between the independent networks in the Internetwork and terrestrial links in a single packet-switching network. Both represent an information transport service supplied without regard to the ultimate application.

The lowest level universal protocol is known as the Internet Protocol, or IP. IP defines a simple service between two communicating hosts on connected networks, attempting only to deliver small units of data called datagrams between the two communicating hosts. IP is run cooperatively by the two hosts and by gateways that link the networks between the hosts. The major functions that IP executes include routing over individual networks, choice of interwork routes (both based upon an Interwork addressing scheme), and datagram fragmentation and reassembly in accordance with the maximum sizes allowed by the networks along a datagram's route.

IP is a military standard and is described in MIL-STD-1777 [MIL1777].

IP does not explicitly provide for reliable data transfer, even though all of the networks between two hosts may be extremely reliable and capable. Instead, IP users must acknowledge that occasional crashes, buffer overflows, queueing delays, etc. may disrupt the flow of datagrams, producing losses, duplicates and out-of-order sequences of datagrams. The principles of IP operation are discussed in greater detail in [POST80b] and [POST81a].

A second, alternative internetwork transport protocol is known as the "stream protocol" (ST) [FORG79]. It resembles IP in that it provides for routing of data across multiple networks, but it differs from IP in that it reserves channels for continuous use instead of receiving and sending datagrams whenever possible. ST is used in conjunction with satellite channels and voice communications and has had limited use to date.

The Internet Control Message Protocol (ICMP) [POST81] defines the exchange of messages for error reporting and similar actions between gateways and IP hosts.

The following describe in part the model of operation of an IP gateway:

1. receive datagrams from one net;
2. determine next net, host to send to (this could be a gateway);
3. fragment the datagram if necessary to meet requirements of next network;
4. send datagram out on network(s).

In addition, a gateway must:

1. multiplex above functions among its networks and internal processes;
2. participate in network management functions--exchange of status and control messages with other gateways and with controller/manager machines; protocols for carrying out this function are discussed below in Section 1.3.4.

1.3.3 Protocols And Techniques For Internetwork End To End Services

The Transmission Control Protocol (TCP) [MIL1778] operates as a "user" of IP to furnish a transparent, reliable data stream service between two communicating hosts. TCP assumes that the networks that carry IP data may be quite unreliable and so takes responsibility for assuring message delivery through positive acknowledgements, timeouts and retransmissions. Furthermore, TCP can multiplex separate data stream connections for different processes running on the same two communicating hosts, so that two users at one host might be simultaneously logged on to the other. TCP runs strictly within the end-user hosts, detecting errors and attempting to correct them through requesting and granting retransmissions. There is no TCP operation executed at gateways between the networks.

TCP is a military standard and is described in MIL-STD-1778.

There are three current major application-style protocols that use TCP:

1. The Telnet protocol which facilitates terminal-to-host communications by handling particular terminal features and operations in a uniform manner. Telnet uses TCP as its lower-level service (I/O channel) [POST83].
2. The File Transfer Protocol which executes file transfers over a TCP connection between hosts and their file management systems [MIL1780].

3. The Simple Mail Transfer Protocol which executes message transfers between hosts and their message handling systems [MIL1781]. Messages and message handling systems are very similar to files and file handling systems, but they are customized to provide a specific "mail service" user interface. SMTP also uses TCP as its network I/O channel.

Applications that are more transaction-like, such as looking up a name in a remote directory, can use an alternative protocol, the "user datagram protocol" (UDP) [POST80a]. UDP supports single, reliable data transmissions without setting up connections and allocation of system resources (as performed by TCP).

1.3.4 Internetwork Organization Concepts

The current doctrine for Internetwork organization defines multiple classes of gateways, including the following:

- o "core" gateways that are operated and controlled by Bolt, Beranek and Newman, Inc.;
- o "exterior" gateways that are not members of the core gateway system.

This organization acknowledges the problems in administration and control over a large number of gateways, and it provides a basis for subdividing the population of gateways. The control problems arise both from the "N-squared" growth of routing update traffic among all gateways under a monolithic organization and from the difficulties of establishing trust among a wide variety of gateway implementations.

In the core gateway system, all gateways do exchange routing information with one another, and they participate in an identical, global routing algorithm. The protocol for accomplishing this is known as the Gateway-to-Gateway Protocol (GGP) [STRAS79]. Core gateways also communicate with exterior gateways using the Exterior Gateway Protocol (EGP; see below) when there is a common network connecting them.

Exterior gateways are constrained either to use a standard information exchange protocol, "Exterior Gateway Protocol" (EGP) among gateways sharing a common network, or else an independently designated interior gateway protocol (an IGP) among a pre-assigned gateway group, termed an "autonomous system." EGP can be used by independent, exterior gateways to have limited participation in the global Internet. Under the EGP doctrine, the independent exterior gateways are constrained to (1) route datagrams either to a neighbor sharing a common network or to a core gateway and (2) to share a common network with a core gateway. EGP [MILLS84] defines the exchange of messages solely between gateways for the purpose of determining limited internet topology. Under EGP, gateways acquire one another as peer neighbors and periodically exchange messages about existence of other peers, and other networks. The range of direct EGP exchanges is limited to gateways that share a common

network. This doctrine does limit the extent of the connectivity information that any one gateway can collect. EGP does not prescribe routing calculations nor distance measures between gateways over networks, except to define a network as a distance of "one hop." To a gateway both the Defense Data Network and a single Ethernet appear as single hops, even though delays may be markedly different.

Collections of gateways may also use an Interior Gateway Protocol of their choosing to perform routing within a select group. For example a collection of gateways could define a protocol identical or similar to GGP for their limited routing algorithm and information exchange. For routing outside of this group, EGP would still be needed. To date, the best known instance of a gateway group running independent IGP's of its choosing and design are those at Bolt, Beranek and Newman's laboratories.

Figure 1-2 illustrates the relations among the above protocols in the DoD internetwork environment.

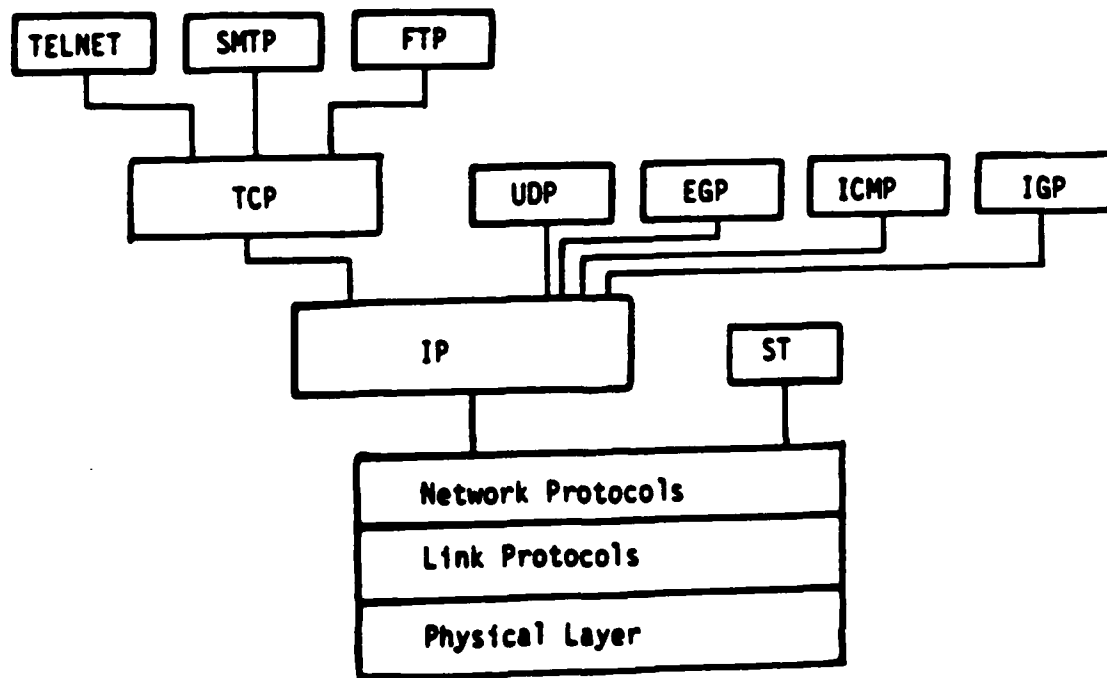


Figure 1-2 Protocols in the Internetwork Environment

1.4 Internet Architecture's Capabilities

The Internetwork architecture used in the Department of Defense has the capability to furnish a data communication virtual channel to any two end user hosts, despite a wide variety and large number of different networks that must be connected. It inflicts minimal requirements on the constituent networks, allowing them to be designed and operated in the best interest of their immediate user communities. This architecture routinely provides application services across multiple networks in a transparent fashion to user applications such as File Transfer and network Mail and remote logons (TELNET). The capabilities, achievements and design issues encountered in the DARPA Internet are discussed by Robert Hinden and colleagues in [HIND83].

1.5 Internet Architecture's Limitations

As experience with actual use of the Internet has accumulated, performance problems have arisen, usually in response to particularly high internetwork resource utilization. Such problems are likely to increase as the Internetwork grows in size and complexity. Consequently there has been a continuing serious examination of real and potential limitations to the internet architecture. For example, [MACOM85] discusses issues inherent in gateway protocols. This process is likely to result in proposed changes to the current internetworking techniques, which will certainly affect future gateway designs. The most serious areas of concern are outlined below:

1. There is a need for more sophisticated resource allocation techniques. Increasing data traffic volumes increase the probability of conflicting resource demands upon gateways and network links. The inherent simplicity of the Internetwork architecture demands that gateways have no "state" information. Without such state information, there is no informed strategy for resource allocation. Therefore, packets are discarded if they cannot be stored. Adverse consequences such as accelerated congestion due to retransmissions are more likely to occur than they might be without more detailed resource allocation techniques.

In addition, there is also a need to match resources to the types of applications using the internet, in the form of "type of service" routing. The three major DoD internetwork applications described above each have different type of service requirements: low delay for Telnet traffic, without regard for bandwidth, and high bandwidth for FTP and SMTP, without regard for delay. Current gateway routing cannot choose very well routes that provide these specific types of service, even though there is a header field in IP datagrams to indicate type of service (TOS) requirements. The consequence is occasional wrong type of service for the user--high delay for Telnet, low bandwidth for FTP and SMTP.

2. There is concern that current addressing strategies, in which the routing information is present in the address (i.e., through a "network" field and a "host" field), will not support such techniques as multicasting, host multi-homing, host mobility and recovery from network partitioning. (See [SUNS82].)
3. Internetwork topologies must be simple and constrained under EGP, because current resource allocation techniques are not sophisticated enough to handle both the administration and control of a very large number of gateways and to prevent resource-consuming routing loops.

The internetwork research and development community is currently considering new techniques for addressing these problems. The techniques under consideration include "logical addressing" in which the network and other route or physical information is not embedded in the address but may be obtained through a table look-up instead, and "flow labeling", in which a gateway would be able to identify and monitor data flows through itself. This would aid in more sophisticated resource management, such as congestion and flow controls. Both of these techniques would represent departures from the current set of internetworking techniques, having impacts upon future gateway designs.

1.6 Internetworking In Non-DoD Environments

There is a mandate for the DoD to be able to use non-DoD networking and internetworking techniques to facilitate wider interoperability and resource availability. However, commercial long-haul networks have evolved along a distinct path from those of the DoD. This section briefly examines major non-DoD internetworking techniques and the potential roles for DoD gateways in use of these techniques.

The development of internetworking techniques using gateways has proceeded less rapidly in other segments of the data communication world, including public and commercial network services. There have been fundamental institutional differences that have influenced the technical directions. First, there is a much higher degree of individual user financial accountability in these latter communities than in the Department of Defense. Each second of holding time for an inter-process connection is chargeable to the user, and there are per-connection establishment costs. As a result, inter-process remote communications tend to be reliable connection oriented rather than based upon a connectionless, potentially unreliable service. Second, the service provided by the public telephone networks is more reliable than that provided by Department of Defense services; therefore, the development of reliable end-to-end protocols (like DoD's TCP) has lagged and has produced multiple classes of a similar protocol (TP Class 4 resembling TCP).

Nevertheless the International Standards Organization and its members have begun addressing techniques for internetworking. Standards exist for linking two X.25 networks (X.25 is the standard family of networking protocols for the operation of channels, links and interactions between a host and a networks)--the X.75 standard. The technique underlying X.75 is to define a connection between two X.25 entities such that the X.25 connections are concatenated. This solves the primary relay and routing problem. The X.75 technique has little of the "end-to-end" flavor of the DoD IP. For error and congestion control, it relies upon the constituent networks forming the X.25 connections. It does not allow for packet reassembly but instead specifies a small maximum packet size of 128 octets. For addressing, X.75 relies upon the X.121 standard for international multinet network naming techniques. (See [CCITTa] and [CCITTb].)

The flexibility of the DoD internetworking architecture can allow use of X.25 networks in a communication path between two end users. To do this, gateways can be configured as hosts on the X.25 net(s) and on 1 or more other participating networks. The gateway's IP process uses the X.25 transport service to send datagrams over the X.25 network. This approach is illustrated in Figure 1.3. Capability to use X.25 networks is a requirement in DoD's major gateway procurements. However, this approach necessitates additional gateway software, in the form of a "connection manager" to send IP datagrams over X.25 virtual connections. The connection manager would need to perform such tasks as establishing and disconnecting virtual circuits, multiplexing datagrams onto existing virtual circuits, deciding upon allowable idle times for a virtual circuit (before disconnecting it), etc.

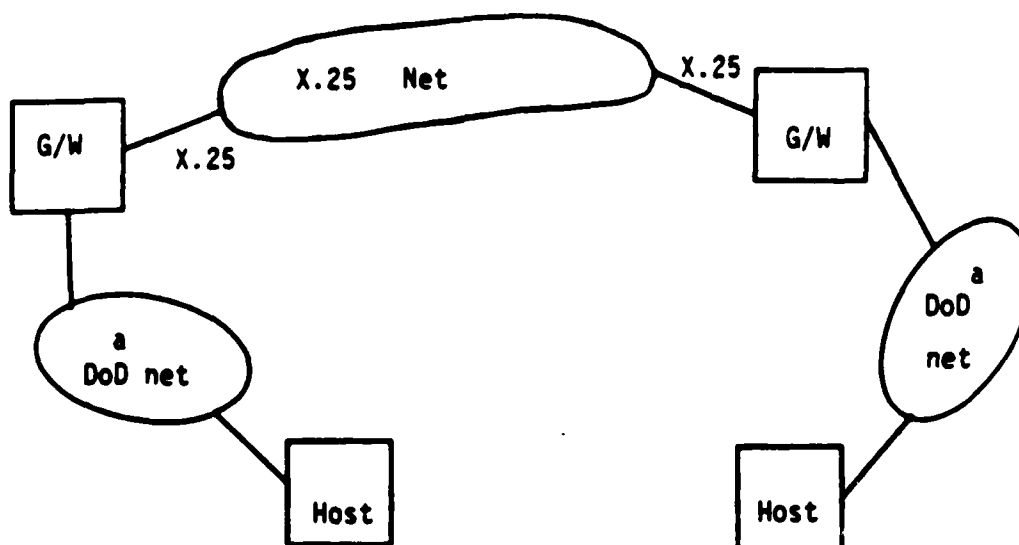


Figure 1-3 Use of X.25 by DoD Internetworking Architecture

1.7 Gateway Mission In Internetwork Architecture

The gateway's mission in the DoD Internetwork architecture is very similar to that of a packet switch in a store-and-forward packet switching network. The gateway must receive and forward internetwork data units (datagrams); it must make routing decisions based upon its information concerning the status of other internetwork resources; and it must participate in network maintenance by measuring, statistics collection and reporting, exchanging connectivity information, and so

forth. These functions are discussed in more detail in Section 3.

2.0 REVIEW OF EXISTING GATEWAY DESIGNS

The section reviews the designs of several existing gateways. Additional sources of review of gateway designs include [FAM83] and [MACOM85].

2.1 The DARPA Internet Gateway

The DARPA Internet gateway was developed by Bolt, Beranek and Newman (BBN), and is described in [HIND82]. It is written in Macro-11 assembly language for a DEC PDP-11 DEC LSI-11 16 bit processor and runs under the Minimal Operating System (MOS) [MATH79]. The DEC 11-series architecture features a single bus system, the "Unibus", supporting a central processing unit (CPU), memory, and devices for controlling serial communication lines, interrupting the CPU, transferring received characters to memory, etc. The memory address width is 16 bits in this architecture, so only 64K bytes can be addressed by the CPU. Consequently, its gateway software was developed in assembly language for maximum space efficiency.

MOS (first developed at SRI) denotes the "Minimal Operating System." It is a collection of service and control routines to support real-time multi-tasking in a dedicated environment. It provides facilities for multiple processes and their scheduling, interprocess communication, buffer management, asynchronous input/output and a shareable real-time clock.

The DARPA Internet gateway implementation is used as a core gateway of the DDN core autonomous system. It runs GGP to determine connectivity to the neighboring gateways inside the core autonomous system, and EGP to determine reachability of networks outside the core system boundaries. This gateway provides forwarding of internet datagrams with IP, and error reporting, routing assistance and flow control with ICMP. It also provides for the collection of status information and monitoring with the Host Monitoring Protocol (HMP), and for restart, recovery, and down load data with the Cross Net Debugger (XNET).

BBN has developed device driver routines to allow the gateway to communicate with the following network interfaces: ARPANET, Ethernet, Packet Radio Network, Proteon Ring Network, RSRE Network, SATNET, and TELENET X.25.

2.2 The MITRE Gateway

The MITRE Corporation is building a prototype low-cost gateway, under contract to the DDN Project Management Office. The goal is to provide both proof of concept and specifications for a minimal hardware DDN external gateway. This single board gateway uses an Intel 80186 microprocessor and is optimized to support only two networks: the MILNET and the Ethernet.

The single Multibus form-factor board contains the 80186 processor, 256K bytes of RAM, 64K bytes of operating system PROM, 64K bytes of user PROM, a Multibus interface, and an internal bus. In the gateway configuration, the Multibus is merely used to provide power to the board.

The MILNET interface is an SBX daughter board of MITRE design, which is used to drive the MILNET IMP via X.25 link level protocols at a rated speed of 100K baud. The Ethernet interface is an Intel 82586 VLSI controller rated at 200-500 packets/second. The 82586 has direct access to the gateway's RAM.

The MITRE gateway software provides the full functionality required of the DDN. It implements the required protocols for an IP exterior gateway: IP, ICMP, and EGP. The directly connected networks are supported via 1822J (HDH) and Ethernet (with ARP) drivers.

The software protocol components are supported by the Excelan NX200 operating system [EXCES4]. This off the shelf, real time operating system resides in kernel PROM. NX200 provides networking facilities similar to the CMOS operating system [STER80], used extensively by MITRE in previous network prototyping efforts. (CMOS is the C-language derivative of the "Minimal Operating System" used in the DARPA Internet Gateway.) These facilities include scheduling, process creation/deletion, interprocess communication, timer management, and I/O support for the Ethernet interface.

The MITRE gateway is designed to work as a single interface between the DDN and an autonomous system. The board will also be used as a building block in the construction of multi-board gateways that provide rich connectivity and robust behavior. This team of gateways will appear as a single logical gateway to the DDN, through the use of logical addressing, and to the hosts, through the use of multicast addressing. Dynamic reconfiguration, load sharing, and a degree of load leveling will be topics of further research.

2.3 The Ford Aerospace Communication Corporation Multinet Gateway

The Ford Aerospace Communication Corporation (FACC) Multinet gateway has been developed under a contract with the Rome Air Development Center (RADC), with efforts beginning in 1981. The gateway has been designed to meet the requirements of classified, critical subscribers, exemplified by the Strategic Air Command. It aims to enable the interfacing of multiple security levels to multiple types of networks, including public data networks. To meet its these aims its design encompasses techniques in computer security and communications security as well as network protocols and hardware reliability.

Given its potentially critical missions of supporting the strategic community, this gateway is designed to operate under human supervision. It supports an operator interface that permits control of its network interfaces and other configuration parameters and allows gateway status monitoring. This feature distinguishes it from most other gateways, which are designed for unattended operation.

The FACC Multinet gateway is developed around a multi-processor hardware architecture. In a typical configuration, several single processor elements, each with memory modules and I/O interface modules, are interconnected via a redundant high-speed signalling bus, termed the "Multiprocessor Intertie Bus" (MPIB). The processing units are Zilog Z8001 16-bit microprocessors. The processor redundancy provides a basis for reliability and high performance. Also, the MPIB provides mechanisms for segregating the processing elements, assuring against the intermixture of classified and unclassified data.

Each processing element, in the form of a set of three boards containing CPU, memory and I/O circuitry, is simultaneously attached to both redundant busses (MPIB-A and MPIB-B). The busses communicate control, address and data information among the separate processing elements at an instantaneous rate of 12 megabytes per second. (That is, the address/data portion of the bus is 2 bytes wide, and the bus operates at about 6 MHz.) The bus performs an address mapping protocol, decomposing composite logical addresses into designations for receiver, message type and priority. (A composite logical address has 12 bits: 2 for priority, 4 for message type and 6 for sender/receiver virtual address.) Therefore, the MPIB defines a well-constrained message-passing medium among the processing elements.

The multi-processor design allows many separate functions to be performed simultaneously. FACC's design has separate processors responsible for the following functions:

1. the IP protocol;
2. gateway monitoring and control functions;

3. network access protocols (i.e., that interact using a network-specific protocol, such as the ARPANET Host-Imp Protocol);
4. link access modules;
5. cryptographic functions;
6. man-machine interface and the "monitor override module";

RADC recently requested bids for the development of an additional processing element to be responsible for gateway routing decisions. This is a potentially compute-intensive functions, and so its relegation to a separate processing element illustrates the benefit of the multi-processor architecture. FACC is likely to be awarded this continued development.

The overall hardware architecture is essential to the gateway's ability to handle both classified and unclassified data and to use encryption techniques to transmit classified data over unclassified networks.

1. The architecture does permit segregation of processing of classified and unclassified data. Communication restrictions between modules can be enforced by the MPIB's capability for resolving composite logical addresses. Under this capability, a software design error is far less likely to result in accidental exchange of information across processor/classification boundaries.
2. The architecture allows integration of encryption/decryption services with the other processing services (e.g. IP, network interfaces, etc.); both a ciphertext processor and a plaintext processor are attached to the MPIB; they are also interconnected through COMSEC equipment; classified data may be prepared by the IP processor and a network access processor, presented to the plaintext processor and then read from the ciphertext processor (in encrypted form) into a network's link access module.

The "C" language has been used for development of multinet gateway software. The following major functional areas are performed by developed software:

1. link protocols;
2. network protocols;
3. security functions such as label checking, encryption, and decryption;

4. IP functions;
5. gateway control functions (e.g. Exterior Gateway Protocol, Internet Control Monitoring Protocol, Routing, etc.)
6. man-machine interface functions;
7. monitor/override functions.

Overall the multinet gateway forms a distributed system, with each processor having limited but rapid communications with the others. Each processing element runs a self-contained software system containing routines for starting from a halted state, self-testing and utility (operating system) functions designed for secure operation. Each processing element also runs its specific functional module(s) (e.g., IP, network access protocols, etc.). The system design allows for the flow of data from a link module, through security/encryption modules, through network interface modules to be recognized as datagrams by the IP module. This sequence is reversed for transmission of datagrams.

Software development has been carried out in a top-down, formal manner with a goal of obtaining an A1 Certification from the DoD Computer Security Center. There are explicit models for the flow of data through the gateway, and there is specific correspondence between security policies for handling those data (e.g. that no process of a higher classification level shall write data to one of a lower level), process and processor allocations, and data flows.

The multinet gateway currently implements standard IP, HMP, ICMP and EGP protocols, exercising the IP security option to distinguish datagrams at multiple security levels. (Its ICMP does not issue "source quench" messages, however.) Its network interfaces include ARPANET (DDN), X.25, SACDIN (using the ADCCP Mode VI [DIN IIa] and Segment Interface Protocol developed for AUTODIN II [DIN IIb]), and the Flexible Intraconnect Local Area Network (a fiber optics based network developed by RADC [MIL-(FI)]).

The multinet gateway is capable of processing 150 datagrams per second when encryption functions are not required, and 50 datagrams per second when using its KG-84 encryption/decryption device in conjunction with the plain and ciphertext processors.

2.4 The Bolt, Beranek And Newman Butterfly Gateway

The Bolt, Beranek and Newman Butterfly (TM) gateway is based upon BBN's Butterfly parallel processor. (Butterfly is a trademark of Bolt, Beranek and Newman, Inc.) This machine is a general-purpose one that was initially developed for DARPA during 1982 and 1983. The Butterfly represents an architecture, rather than a single machine, because its components are modular and be tailored into many different configurations. The fundamental processing unit, termed a "Processor

Node", is a single board Motorola 68000-based computer with I/O and memory interfaces. From one to 256 such Processor Nodes can be assembled into an active machine.

The Butterfly is not solely intended for use as a gateway. It is now represented by its manufacturer as a general purpose architecture for evaluation around several types of applications that lent themselves to parallel processing: multi-dimensional signal processing, simulations, scientific (especially matrix) computations, design rule processing for VLSI, molecular modelling and image processing. For applications that can be rendered into parallel tasks, the Butterfly's processing rate has been estimated at 60 million instructions per second (60 MIPS) for a machine with 128 Processor Nodes [ROSE85]. Based upon the potential for efficient execution of gateway tasks on parallel processors and upon the need for extended processing resources for DARPA core gateways, the Butterfly will be used to replace core gateway machines. (The core gateway machine is discussed in Section 2.1.) As a gateway, the Butterfly can run in an unattended environment, supplying monitor and control information (as a network host) to a Network Operations Center.

The "Processor Node" is the unit of both computational and memory resources for the Butterfly. Physically, it is a 12 by 18 inch printed circuit board containing a Motorola 68000 microprocessor (running at 8 MHz), main memory, memory management hardware, an I/O bus, an Advanced Micro Devices 2901 based co-processor (termed a "Processor Node Controller"), and an interface to the Butterfly switch (the inter-processor interconnection network).

A single processor node is capable of extensive I/O operations through connection to up to four I/O boards and Multibus (TM) [IEEE83] connections to other devices operating Multibus protocols. The I/O bus is 16 bits wide and can accommodate up to one million data transfer operations per second.

Application processes running on a Processor Node "see" a single virtual address space. Specific memory locations are mapped into I/O devices and memory that is local to other Processor Nodes, as well as to memory that is resident at the same Processor Node. The memory management unit (MMU) is responsible for resolving virtual memory references (as coded into an application program) into their physical references. The Processor Node Controller actively uses the MMU to resolve non-local memory references, which could be to I/O devices, or to memory locations on other Processor Nodes.

The local memory for a single Processor Node is 256K bytes when the sockets are populated with 64K RAM chips, or one megabyte when the sockets are populated with 256K RAM chips. Daughter boards can be used to expand memory up to 4 megabytes. EPROM is used to store a bootstrap loader, power-on diagnostics and a debugger.

The Processor Node Controller (PNC) and the switch interface provide intelligence necessary to carry out inter-Node memory references and transfers. (The Processor Node Controller also governs all control signals necessary to carry out local data transfers, and it performs process synchronization and queueing primitive operations on behalf of the operating system.) Its data paths are 16 bits wide to match to M68000 data bus. It operates from microinstructions stored in a 4K by 64 bit control store, at a rate of 8 million microinstructions per second. When an external memory request is made by an application program, the PNC determines the external character of the request and places a read request containing a processor number and remote memory location to the switch interface. The PNC awaits a reply but may service other micro-interrupts (e.g., to service a remote access from another processor node). The M68000 simply waits. When the reply arrives, the PNC hands it to the M68000 as though a local memory request had occurred. The access delay for remote memory is 4 microseconds, much longer than that required for local memory access. However, the PNC can also be instructed to move blocks of locations, at rates up to 32M bits per second.

The actual Butterfly switch is implemented as either a board populated by medium-scale integration (MSI) devices to support a four-by-four cross-bar switch or by a board populated by VLSI devices to implement a 16-by-16 switch. (The 16-by-16 switch is not in itself a cross bar switch. Consequently, data may need to be routed/recycled through a 16-by-16 switch in order to reach a desired processor node.) The switch logic is responsible for routine, timing, flow control and collision resolutions to send packets through a single switch [GOODE85].

The Butterfly gateway software, including the operating system, "Chrysalis" (TM) has been written in the C programming language. (Earlier BBN-developed gateways used MACRO-11 assembly language to conserve memory space.) The Chrysalis operating system supports process management and communications as well as buffer management. Process images are initially loaded in run an external device and then controlled by a scheduler. Processes may create and run other processes on other processors by first copying the necessary memory images to the target processor via the Butterfly switch. Chrysalis queueing, scheduling, event and object management support synchronization of multiple processes working on a common task, pipelining of a task among different software processes, etc.

Currently, the following protocols and network interfaces have been implemented for the Butterfly gateway:

1. the DoD Internet Protocol [MIL1777];
2. Internet Control Message Protocol (ICMP) [POST81];
3. Exterior Gateway Protocol (EGP) [MILLS84];

4. Host Monitoring Protocol (HMP) [LITT81];
5. Reliable Data Protocol (RDP) [VELT84];
6. ARPANET 1822 Interface (AHIP) [BBN78];
7. Ethernet Interface (see [IEEE802]);
8. X.25 [CCITTa] and [CCITTb];

The performance rates of the Butterfly gateway are noted by BBN as follows:

- 1 Processor : 100 packets per second
- 4 Processors: 500 packets per second
- 8 Processors: 1000 packets per second
- 16 Processors: 3000 packets per second.

2.5 EGP Under Berkeley 4.2 UNIX (DEC VAX)

The EGP gateway under Berkeley UNIX 4.2, also called ISI-Hobgoblin, was developed by the Information Sciences Institute (ISI). ISI-Hobgoblin runs on a DEC VAX (TM) 11/750 32 bit processor. The VAX 11/750 processor supports a multi-user multi-tasking environment. This machine is not dedicated to gateway functions, but also supports a community of regular users. This is unusual for gateways.

(VAX is a trademark of Digital Equipment Corporation. UNIX is a trademark of Bell Laboratories.)

(Although a machine of the 11/750's capability is "overkill" for gateway functions, the ISI-Hobgoblin is an interesting application of computing power, also demonstrating that it is not always necessary to house a gateway in a dedicated microprocessor "box." On the other hand, it should not be assumed that a VAX 11/750 is required for building a gateway.)

The VAX hardware architecture is a well-known, highly capable general purpose one that features a large, powerful instruction set and a virtual memory system. Thus on the order of 10 user processes, each having access to the full 32-bit address space. (However, much of that address space is reserved for operating system routines and use.)

The software is written in C language and runs under the Berkeley UNIX 4.2 operating system. This is the current version of UNIX that supports the standard DoD network protocols. UNIX is an operating system which supports a real-time environment. It provides functions to support virtual memory, interprocess communication, and I/O functions.

At the user level, it permits the interactive programmer to access, control and debug the application software.

The motivation for ISI to build this gateway was the current DDN topological restrictions that do not allow any non-routing or dumb gateway connected to the DDN core autonomous system, as explained below. ISI-Hobgoblin is a host connected to the ARPANET and to the ISI-network. The University of California at Irvine network (UCI-ICS) is connected to the ISI-NET through a non-routing gateway. ISI-Hobgoblin runs EGP to inform the ISI-gateway, which is a core gateway, of the accessibility of the UCI-ICS network and to determine the connectivity of other networks outside its own system boundaries, complying with the DDN requirement of informing the core of exterior autonomous systems through an exterior gateway.

Besides the EGP, ISI-Hobgoblin implements the standard set of gateway protocols IP and ICMP. The ICMP redirect message is not yet implemented but will be added later. This type of message is important to assist to the effective routing of datagrams.

There is no current implementation of an IGP in the ISI-Hobgoblin. The information about the connectivity of interior network UCI-ICS and the interior gateway is provided at initialization time.

2.6 SDC-Burroughs Gateway

The SDC/Burroughs gateway was developed by System Development Corporation as part of their MIL/INT product line. There are two versions of this gateway each of which connect two networks together, and the only difference is the type of network interfaces. Each runs IP, ICMP, and GGP to provide the basic functions of a gateway.

The gateway architecture is composed of two Intel 8086 microprocessors. Each processor contains its own (local) ROM and RAM. The two processors also have access to a common memory unit. In current configurations, each processor contains a total of 128 K bytes of memory. The processors' physical configurations are identical, but each runs function-specific software. A proprietary operating system manages all of the software functions. One of the two boards can be and is arbitrarily designated as the master and runs IP, GGP and ICMP. Each processor also contains protocols to control modems and execute one network interface driver (e.g. CSMA/CD or ARPANET Host-IMP Protocol--AHIP [BBN78]). Each processor occupies a single board that fits into a Multibus backplane, permitting both global and local communication and control paths. The Intel Multibus (TM) is the inter-processor communication standard. ("Multibus" is a trademark of Intel Corporation.)

Two configurations are currently available:

1. a version to allow interconnection between a DDN IMP and the SDC local network (a Broadband CSMA/CD based cable network);
2. a version to allow interconnection between distinct channels on the SDC broadband local network.

Two means are available for software configuration management. The gateway may be configured with EEPROM so that operational code is available immediate after power-up. The gateway may also be down-loaded using a smaller loader within a kernel residing in ROM, so that operation software may be changed at will.

SDC-Burroughs is planning improvements for future versions of the gateway, including an X.25 network interface, expansion of the processor memory complement to 512 K bytes, support for Ethernet interfaces, and development under a larger processor, such as the 80186.

2.7 The SAC Gateway

The Strategic Air Command (SAC) Gateway System is an integral part of the SAC Internet system under development by SRI International for Rome Air Development Center and Defense Advanced Research Projects Agency. The SAC Internet is an experimental system extending the DARPA Internet for internetwork accommodation of network dynamics: mobile hosts, partitioning and coalescing of networks as well as of the internet itself. It consists of the SAC Gateway System and the Survivable Name Service System. It is designed to be interoperable with DARPA Internet across EGP as an independent autonomous system. To date, laboratory demonstrations have been conducted, and a field test is planned for early 1986.

The SAC gateway is written in C, aiming at both LSI 11 and 68000-based hardware. Its current implementation interfaces ARPANET, packet radio networks, and Ethernets. There are three components to the internetworking software: the Host-to-Gateway Protocol (HGP), the SAC Incremental Gateway Protocol (SGP), and the Forward Request Protocol (FRP).

The HGP facilitates host-gateway affiliation and detection of changes in their connectivity. Each host is affiliated with a local gateway. The internet address of a host is "gateway.host", a concatenation of the gateway and a local host ID. The HGP fashions a periodic host-to-gateway echo/reply message pair.

The SGP is a gateway-to-gateway protocol exchanging periodic updates between adjacent gateways. It is similar to the GGP (Gateway-to-Gateway Protocol) of DARPA Internet. The differences stem mostly from the gateway-centric addressing and that each SAC gateway is divided into gateway halves, one each interface. Routing is thus keyed

by destination gateway half rather than destination network. Limiting the knowledge of individual gateways to immediate surrounding curtails effect of network dynamics.

The system architecture adopts a gateway-centric addressing scheme. The address provides the host number and the number of a gateway with which the host is affiliated, rather than the number of the network with which the host is affiliated. If the network is partitioned under a network-centric addressing scheme, recovery is more difficult due to the need to identify two or more partitions of the network. This is avoided in gateway-centric addressing. The gateway-centric addressing confines the potential results of network dynamics to changing addresses. For routing, an incremental algorithm is adopted to minimize the impact of dynamic topological changes. Each gateway maintains knowledge of its immediate surroundings.

The SAC gateway system design is predicated upon reliable transport connections as furnished by TCP. The SAC gateway system design aims at maintaining TCP connections under continuous movements, failures, and introduction of hosts, gateways and/or communication links. No inherent design limit is imposed on the rate of internet changes, except that TCP timeouts may occur during intervals in which movements, failures and new element arrivals occur.

Due to network dynamics, it may become necessary for a host to reaffiliate itself with a different gateway, thus a change of address. Traffic bound for this host on an existing (TCP) connection would be forwarded via a forwarding path set up using the FRP.

The SAC Gateway system implements other standard Internet protocols namely, the ICMP, and EGP. It requires minor modifications to host IP modules, but none for TCP implementations.

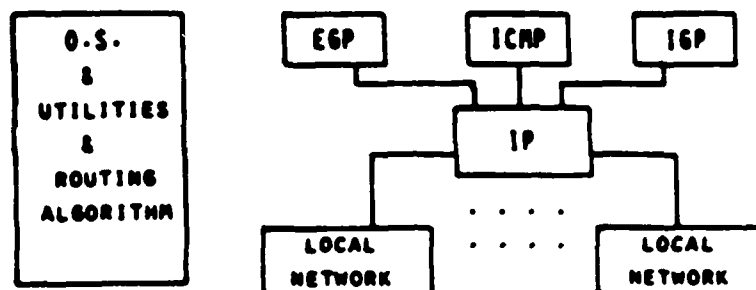
3.0 GATEWAY FUNCTIONAL REQUIREMENTS AND PROTOCOLS

Gateway functional requirements and protocols are the minimal set of capabilities necessary for it to successfully operate in the DoD Internetwork. Hardware requirements for the central processing unit, primary and secondary memory are not as firm. They are dictated more by the types of networks to which the gateway will be attached. This section addresses software and hardware requirements in turn. In addition, it briefly reviews operational requirements.

3.1 Software Requirements

The following software components are required for gateway capabilities:

1. Device drivers to operate data communication via attached networks and to provide "lower level" services in accordance with the requirements for IP [MIL1777].
2. Process to run IP, including the gateway relay function. Mapping between IP addresses and local network addresses is required.
3. Process to run gateway routing information protocols, in particular EGP, but also interior gateway protocols as established by the user.
4. Process to run ICMP.
5. Processes to participate in network management, including network host monitoring, internetwork resource monitoring, code up- and down-loading.
6. A real time operating system to handle communication interrupts, time sharing between different processes, configuration management, and other functions as defined by the user.



EGP - EXTERIOR GATEWAY PROTOCOL

ICMP - INTERNET CONTROL MESSAGE PROTOCOL

IP - INTERNET PROTOCOL

IGP - INTERIOR GATEWAY PROTOCOL

Figure 3-1 Gateway Software System Architecture

Together these software components comprise a time sharing environment to provide time slices for the execution of required gateway protocol functions:

1. IP to relay datagrams, performing routing decisions and fragmentations as required in accordance with MIL-STD-1777;
2. ICMP to respond to conditions requiring message generation and reception in accordance with RFC 792, "Internet Control Message Protocol" [POST81];
3. EGP to exchange status and connectivity information with neighbor gateways in accordance with RFC 904 [MILLS84];
4. processes to maintain databases based on received messages;
5. processes to run host monitoring protocols, e.g. [LITT81].

3.2 Hardware Functional Requirements

A gateway's hardware architecture must be adapted to the particular network functional and traffic volume requirements associated with its intended position in the internetwork topology. Past experience has demonstrated that a variety of CPU types provide useful implementation bases. On the other hand, network interfacing requires special purpose controller (e.g., for an ARPA AHIP (1822) interface, or for an IEEE 802 interface). The following observations summarize principles of gateway hardware architecture selection.

1. Since the gateway is a component of a communication channel, its throughput should not be lower than the sum of the sustained throughput rates of its attached networks. For example, a gateway simultaneously attached to two 1 megabit-per-second cable networks should itself be able to sustain rates of 1 megabits per second. For datagram sizes of 8000 bits, this is 125 packets per second.
2. Hardware peripheral processors should be used to handle network operating functions at rates required by those networks. For example, there are IEEE 802 controller units, ARPANET controller units, and the like capable of direct memory transfers. On the other hand, relegating network interface functions to a CPU would seriously degrade overall packet-handling capacity.
3. A primary determinant of a gateway throughput is its packet handling speed. That is, a context change and processing are required for each newly arrived packet. This has been observed to be around 1000 per second maximum for most implementations, due in part to context change limitations, and due in part to the number of instructions (estimated at 500) needed to process a packet.
4. A secondary determinant of gateway throughput is its memory bandwidth--the rate at which information can be written to or read from primary memory. For example, if the memory access time is 250 nanoseconds, and the memory word width is 16 bits, then data can be retrieved from or written to the memory at rates of up to 64

megabits per second.

A gateway's processor architecture, in terms of its peripheral hardware, its interrupt processing speed and its primary memory bandwidth, must be sufficient to process data arriving from its attached networks.

3.3 Basic Operational Requirements

A gateway must be designed and configured in accordance with its operational requirements. In a non-critical, unclassified environment, these operational requirements may resemble those of the DDN Network Access Component [NAC84]:

1. Unattended operation;
2. Initialization upon power up without aid of mass storage units or network controllers;
3. Extended run periods limited only by inherent hardware reliability, preventive maintenance intervals and constraints of operating environment.

In environments that serve critical applications, such as strategic readiness, stricter requirements are in order. A human operator must be able to perform health checks, configuration changes and the like at any time. Sections 5 and 6 discuss the implication of anticipated DoD gateway operational requirements upon the specific areas of monitor and control capabilities and reliability and maintenance. Section 4.10 discusses the tradeoffs relating to operational requirement alternatives.

Handling classified information and providing service to classified subscribers is a separate and very significant operational requirement whose implications are discussed in Section 7.

4.0 GATEWAY DESIGN TRADEOFFS

The variety of gateway designs makes it clear that the major gateway functional and performance requirements can be met in many ways, creating tradeoffs. This section examines those tradeoff areas, noting the costs and benefits associated with tradeoff decisions. The areas of tradeoff decisions discussed include the following:

1. Local and Remote Monitoring/Control Capabilities.

2. Network Interfaces
3. Enhanced Services for Applications
4. Software Configuration Flexibility Versus Simplicity
5. Interior Gateway Protocols and Routing Techniques
6. Gateway Architecture
7. Gateway and Internetwork Survivability
8. Gateway Physical Characteristics
9. Gateway Reliability and Maintainability
10. Gateway Operational Controls

4.1 Local And Remote Monitor/Control Capabilities

Although gateways can run unattended for long periods, it is frequently necessary to assess the status of a running gateway, for example in diagnosing internetwork problems or to maintain a high level of assurance about the gateway. It is also frequently necessary to collect statistics from the gateway in order to plan future network and internetwork capacity allocations. Several baseline standards for gateway monitoring and control functions exist as implemented in actual machines. These baseline standards meet differing operation requirements and illustrate potential tradeoffs. The existing baselines include the following:

1. Perform remote monitoring using only the capabilities inherent in the gateway protocol suite, in accordance with RFC 904, RFC 792 and MIL-STD-1777.
2. Perform remote monitoring and control using capabilities as defined by HMP and XNET, as described in RFC 799.
3. Perform local monitoring and control using capabilities as defined by the FACC Multinet Gateway Man Machine Interface.
4. Perform local and remote monitoring and control using the capabilities of the UNIX (TM) operating system over gateway processes, as exemplified by the BSD 4.2 gateway.

The capabilities of these techniques are discussed below. They represent a spectrum of management activeness, ranging from passive use of existing protocols to remote management to onsite management.

4.1.1 Monitor Capabilities Of Baseline Protocol Suite

When a gateway runs IP, ICMP and EGP, it is already equipped to recover from network faults. ICMP and EGP define means for communicating with other gateways and hosts, and they provide the means for sensing status changes. The status information is made available to IP in the form of routing tables. When an event such as a network or gateway failure does happen, the recovery will not be instantaneous but will require seconds and minutes as gateways exchange information. (In fact, the necessary recovery time may be beyond the bounds normally allowed for TCP acknowledgements, causing TCP connections to break. The DoD Internetwork R and D community is seeking means to reduce this occurrence through faster gateway updates, but without undue overhead.)

There are actions that monitoring hosts can take within the standard internetwork protocol suite to provide assurances that specific gateways are functional:

1. In the MILNET or ARPANET environment, use of AHIP messages about the status of another network host/gateway;
2. Continuous polling via ICMP "echo" messages;
3. Responsive polling, perhaps to timeout events as detected by TCP.

4.1.2 Host Monitoring Protocol And XNET

The current de facto standards for gateway monitoring and control have been established by BBN for their internet gateway core population. Their gateways run two processes, called "Host Monitoring Protocol" (HMP) and "Cross-Net Protocol" (XNET). The former allows the gateway to report its status as a network host to the internet monitoring center. The latter allows software resident in the gateway to be uploaded, downloaded, started, restarted and debugged. XNET provides a remote agent a very powerful capability to alter gateway operation; it is not suitable for all operation environments.

HMP includes the following information exchanges between a gateway and a monitoring center:

1. Gateway status (UP/DOWN) information. This include the time that the gateway is been up, memory allocation information, number of interfaces, and number of neighbors.
2. Interface status information. For every gateway's interface, the status of the interface, the address, and the buffer allocation is informed.

3. Neighbors status information. For every neighbor of the polled gateway the status and the address of each neighbor is provided to the monitoring center.
4. Gateway statistics. The number of packets dropped due to unreachable networks or unreachable hosts. Notice that this count does not include the overall number of packets dropped. There are other reasons to stop processing a packet such as time-to-live of the packet is zero.
5. Interface statistics. The gateway must provide, for each interface, the number of: packets dropped, IP errors, datagrams for the gateway, and the forwarded datagrams.
6. Neighbors statistics. For each neighbor, the gateway counts the routing updates received/sent from/to that neighbor. Also, the number of packets received/forwarded through/to that neighbor is declared.
7. Traffic flow through the gateway. The gateway has to maintain a Host Traffic Matrix which contains an entry for each (reachable) source/destination pair of the number of packets received or sent. It informs about one or more source/destination pairs at one time.
8. Gateway routing information. For each reachable network, the gateway provides the distance to the network and the neighbor through which that network is accessible.

XNET includes the following types of exchange between a gateway and a monitoring center:

1. load specified data into a specified region of the gateway memory; this allows new code to be tested via placement and execution in the target environment, and it allows remote software configuration changes;
2. read a specified area of the gateway memory; this allows either data or executable code to be remotely inspected during trouble shooting or maintenance operations;
3. execute code in the gateway under the control of a debugging process that allows for single stepping and break-points; this supports detailed remote debugging activities.

4.1.3 FACC Multinet Gateway Local Monitoring

The Ford Aerospace Multinet gateway runs a process whose task is to interact with a locally attached system manager console. This gateway affords the local system administrator the following capabilities:

1. Enter commands at the console and display the requested information. These commands enable the operator to execute the next capabilities.
2. Control the status of the network interfaces. The operator is capable of set and reset the I/O ports to down or ready state respectively, and is also able to read the current state of the interfaces.
3. Initialize the gateway from a cold start and bring it up to operational state.
4. Obtain gateway information such as the status of the gateway, the collected statistics of the gateway, and the gateway's routing data.
5. Obtain classified information only through an identification procedure. This is the case when different security levels of data are handled in the system.
6. Get the attention of the operator when an error occurs in the system. This capability may be implemented by ringing bells and printing many asterisks along with the contents of the error message when an abnormal situation is encountered.

4.1.4 Capabilities Of Internetwork Operation Centers

An internetwork operation center (INOC) must be based around one or more special-purpose network hosts. Such hosts have standard network and interwork addresses, and they run standard network access protocols along with internet protocols. They are special because they run processes that communicate with control and monitor processes in gateways. The center's monitoring/control process operates on behalf of users, collecting information from correspondent gateways, displaying this information, and sending commands initiated by the user to the correspondent gateways. Figure 4.1-1 illustrate these relationships.

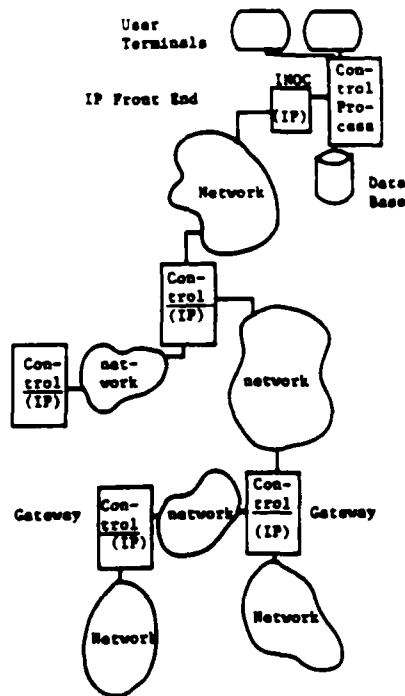


Figure 4.1-1 Gateway/INOC Relationship

Considerable administrative coordination is required to establish this relationship among the INOC host and its particular gateways. The INOC host must be provided with up-to-date lists of gateways to be controlled, and the gateways must be provided with the identity of the host that will control them. Both gateways and the INOC host must be provided with protection mechanisms that guarantee authenticity and integrity of the control messages.

The primary functions of current internetwork operation centers, as exemplified by the one operated by Bolt, Beranek and Newman are to observe the status of internet gateways and their networks and to issue commands (e.g. to close a network interface, or to halt a specific gateway). (These gateways are limited to the core collection that run the BBN-implemented Host Monitoring Protocol.) The INOC host automates the requisite network communications and provides interfaces to the human users responsible for network maintenance. The human users are responsible for the diagnoses and actions taken in order to manage operations and respond to problems in the internetwork.

The size and capability of an INOC host will depend upon both the number of gateways to be controlled, the frequency with which monitor updates and control actions occur, and the number of users to be supported. A minicomputer, such as a PDP 11/70 or VAX (TM) 11/750 is sufficient to support the control of 10-20 gateways and 3-4 user/analysts. Such a machine does not need a large amount of primary storage but does need the capability to support multiple user processes simultaneously and to support a moderate volume of network traffic. Its secondary storage capacity (e.g., disk drives) must be sufficient to support logging requirements. A disk should be capable of storing 1 to 5 days' activities, and tape storage should be available for permanent logging from disk records.

4.1.5 Future Internet Management And Control Techniques

Future techniques for internet monitoring and control must be aimed at automating the diagnosis and remedy procedures that are currently performed by human operators and analysts. This will be necessary both to manage a large population of gateways as well as to provide the best possible management of smaller groups of gateways (e.g., administered as autonomous systems by an independent DoD agency). The existing methods as developed for BBN gateways and for the FACC Multinet gateway do provide the means for gathering information and for disseminating commands, but they do not provide support for diagnosis of problems and for choosing actions in response to problems.

Artificial intelligence (AI) concepts may provide the means for automating the diagnosis and repair decisions associated with managing an internetwork. To put AI to use in this task involves capturing internet management expertise in a form suitable for automated use, such as an expert system. An expert system for internetwork management assistance should have the capabilities to recognize many types of network problem syndromes and should be able to recommend actions to take with respect to the probable failure causes. It should also be able to "learn" from experience gained in manual network management--new fault syndromes and their remedies. For example, it may be possible for an expert system to diagnose a gateway sending faulty update messages by examining a series of individual gateway routing tables.

AI techniques are currently under investigation, and indications are that they can provide significant support in this area. J. Wescott and her colleagues at Bolt, Beranek and Newman [WES85] have reported on the design of a highly automated network management system. Many AI researchers feel that if a cognitive task can be performed by a human expert on a principled basis, then it is possible to automate the performance of that task using rule-based and related AI techniques. On this basis, automation of Internet problem diagnosis and remedy seems feasible. Experts also point to the benefits of machine performance of management tasks--machines are less subject to fatigue and stress than are humans.

The incorporation of techniques for automated network management, including AI-based internetwork management, may require processing power beyond that currently used at INOCs. The actual amount of additional power will be determined by the frequency of diagnosis tasks and the volume of knowledge employed by the automated management techniques. Conceivably, a dedicated minicomputer (e.g. a DEC VAX (TM) configured with on the order of 1 Mbytes of primary memory and on the order of 100 Mbytes of secondary storage) could easily meet continuous automated management requirements for on the order of twenty gateways. However, this is only an estimate. Potential internet managers might be better served by specifying the volume of manual and automated internet management activities to be handled rather than the actual configuration of a machine to handle them.

4.1.6 Tradeoff Discussion

The fundamental tradeoff in the area of local and remote monitor and control capabilities is need versus cost. Given the sophistication of computer operating systems in providing internal health checks and fault tolerant operations, technical feasibility of advanced capabilities not is an issue in itself. Instead, the cost of these capabilities as balanced against their need in finite budgets is an issue. The potential gateway owner must consider the degree to which one or more gateways must be managed in order to support operational requirements and to adapt to their environment(s). For example, if a gateway is to be maintained on-site, then a debugging process need not be coupled with an internetwork communication protocol. Instead, it can be coupled with a simple terminal interface. Major determinants in such a decision will be (1) mission criticality and the necessity for continuous reliable gateway operation, and (2) system complexity--operation of a multi-gateway autonomous system versus a single, local gateway. Tradeoff choices will include the number of items that can be monitored and controlled through a remote or local interface.

The cost versus need tradeoff applies also to remote versus local monitoring and control capabilities. In a highly critical application environment (e.g., in which strategic alertness is supported) local monitoring of the gateway can give added assurance of overall connectivity. In general the same capabilities for monitoring and

control are easier to implement locally through a man-machine interface, compared to remote implementation over the internetwork. (There is reduced software and testing complexity associated with writing to a terminal handling rather than a network handling process.) On the other hand, manned gateway sites are more expensive than unmanned ones.

4.2 Network Interfaces

Gateway design requirements will be driven by both present and future network connection requirements. The most common requirement is likely to be simultaneous connection to the Defense Data Network's MILNET via an X.25 or AHIP protocol and to a local area network such as an IEEE 802 type network. Future requirements could include further network connections of still other network types (e.g. a packet radio network).

Again, the fundamental tradeoff is in cost versus capabilities. Techniques currently exist for connecting machines as hosts to a wide variety of networks, so that technical risk is not an issue. Instead, the potential gateway owner must weigh present needs, future needs and the value of planning for requirements against the cost of the hardware and software to support specific network interfaces.

A safe approach would be to formulate first a general-purpose interface definition between the IP and attached networks as a development standard. This would support easy reconfigurations to allow gateway connection to different networks as the need arose.

A second tradeoff consideration is the impact of multiple network interconnections upon gateway performance. This can be expected both when the attached networks offer more bits per second than can be handled by the gateway processor and when the attached networks are of differing speeds, so that the slower one cannot handle traffic offered by the faster one. The former problem can be addressed through advances in processor architecture. For example, multiple processor architectures have been used to enhance gateway throughput. The latter problem is one inherent in the internet architectural model, requiring a "stateless" gateway. (The immediate consequence of the speed mismatch is the buffering and delay of packets in the gateway. Retransmissions in response to this condition exacerbate the problem, and the gateway has no means to control the problem other than "source quench" messages of limited utility.)

4.3 Enhanced Services For Applications

There are currently some limited opportunities by which gateways could offer enhanced services to applications. The most obvious of these involve the use of IP header fields for specifying precedence, type-of-service and security levels. IP header fields are currently defined for identification of up to 8 levels of precedence, preference for throughput versus delay from a network, and standard military classification levels. In principle, use of these fields could allow internetwork processes to furnish a full military message service and support real time applications. A major motivation for enhanced or at least differing grades and types of service comes from the differing needs of applications. For example, mail service can tolerate minute-long delays, while interactive connections cannot.

There are also current barriers to realizing benefits of these mechanisms, most of which can be traced to a need for simplicity in protocol implementations. Both SMTP (for mail) and TELNET (for interactive connections) use TCP as a transport service, which in turn uses IP. This separation of applications from IP precludes differentiation of their service requirements. As another example, the IP type of service field can be used for requesting network-specific services. In principle the type of service field could also be used in routing datagrams with respect to their throughput-versus-delay and accuracy requirements by directing that the routing algorithm use either throughput-oriented or delay-oriented objectives.

Although precedence levels are specified for IP datagrams, few current gateways are capable of distinguishing and enforcing the precedence levels. Those that do have been constructed in response to the needs of special communities such as SACDIN and DODIIS. The addition of true precedence handling capability requires major augmentation of a gateway operating system in the form of a multi-queue service. However, future gateways will require precedence handling of datagrams to ensure continuing service to high priority applications under degraded operating conditions.

The choices faced in specifying a gateway include the following:

1. Furnish only required precedence services without type of service distinctions.
2. Provide mechanisms for precedence and type of service, and assist an informed user community in the direct use of the latter features for their applications.
3. Design a means for the gateway to distinguish applications and their requirements and then intelligently differentiate and fulfill their type and grade of service requirements.

4. Rely upon a substitute for IP when the application requirements are significantly different and extreme, as in the case of the Stream protocol ST, for voice applications, that reserves resources.

One through three represent an increasingly expensive series of approaches. The first represents many existing gateways. The second has been criticized because of the sensitivity it would produce with respect to users' possibly poor choices (e.g., a bulk transport application requesting highest types and grades of service). The third is regarded as requiring very complex software and capable processing to continuously optimize resource allocation. The third might also require violation of the protocol layering principle so that a gateway could discern datagrams' requirements. However, it does offer potential protection from users' poor choices.

4.4 Software Configuration Flexibility Versus Simplicity

Ideally, machines such as gateways that furnish network access can run with a minimum of human oversight and they can use read only memory rather than magnetic disk or tape as their non-volatile storage media. The disadvantages of a ROM-based approach are realized when frequent updates to the executable code become necessary. The set of core gateways maintained by Bolt, Beranek and Newman represent yet another approach that avoids this difficulty. These gateways can be down-loaded and controlled over the network. Clearly there is significant expense and development associated with this approach to gateway software configuration management. Disadvantages of magnetic storage of executable code include the necessity to reload on cold and warm starts and the added expense of mechanical equipment (i.e., disk drives) and software to manage them.

Specific requirements and their corresponding techniques for executable code storage and configuration maintenance are shown below.

<u>Requirement</u>	<u>Memory Medium</u>
Low expense; short simple power-up sequence	ROM
Frequent Software Updates	Magnetic Media (Tape, Disk)
Update software Frequently for Internetwork System Testing	ROM Kernel plus Network Downloading

4.5 Interior Gateway Protocols And Routing Algorithms

Gateways need to exchange status information periodically to permit the internetwork adaptive algorithms to respond to network configuration changes and other problems and bottlenecks. Therefore, inter-gateway messages are needed to communicate a variety of conditions: the existence of neighbor gateways that a sender gateway is aware of, the existence of networks that the sender gateway is a member of and knows to be operational, etc. These messages, when exchanged among gateways, allow each to build and update its own "model" of the internetwork topology. The model is subsequently used for IP routing decisions.

The current doctrine for the DoD Internetwork organization defines three types of inter-gateway information exchanges:

1. (core) Gateway to (core) Gateway Protocol (GGP), described in [STRAS79];
2. Exterior Gateway Protocol (EGP); and
3. Interior Gateway Protocol.

The middle one of these defines only a method of data exchange, and not a routing algorithm among gateways. The other two protocols also allow for definition of routing functions. In the case of gateways implemented outside of the BBN core system, implementors have freedom to design gateway routing procedures.

The major tradeoff with respect to gateway routing places the difficulty of achieving metrics for delay and throughput against the need for optimal use of resources. It has been difficult to achieve good measurements of network delay, because it is both difficult to accurately keep network time and difficult to measure precisely the network delay rather than host processing delays. (See [MILLS83] for discussion of Internet delay experiments.) However, the issue of separating network from host delays is inherent in the layered approach to protocol implementations and to system implementations in general. An IP process can periodically read the universal network time, synchronizing a local clock, and then use this information to time-stamp datagrams. The datagrams are queued to a device driver, and the time waiting in the queue becomes part of the perceived delay! On the other hand, accurate time measurement is an absolute requirement for measurement of network throughput as well as delay and for resulting resource optimizations. The Internetwork R and D community recognizes the absence of methods for accurate delay measurements and the need for a standard method.

A second tradeoff involves the difficulty of close control of internetwork resources against their benefits. The current approaches to Internetworking are based upon loose control, but these have actually developed from an earlier, more tightly controlled configuration. The initial population of gateways in the DoD/DARPA internetwork all ran a common gateway-to-gateway protocol (GGP), and they performed best route calculations using a common algorithm. As more gateways were

implemented, inherent difficulties in the administration, enforcement and trust of multiple gateways were recognized. The exterior gateway protocol was developed as a means of coordinating activities of mutually suspicious gateways. EGP effectively divides gateways into communities of trust, based upon physical proximity and administrative considerations. Mutually trusting groups of gateways are termed "autonomous systems", and they can exchange EGP updates. A core gateway system under BBN control is regarded as trusted by every autonomous system. Currently, the internet R and D community is contemplating extensions of the underlying assumptions for exterior gateways.

The tradeoff posed for the potential gateway owner involves the acceptance of the current limitations of the EGP gateway (by specifying EGP gateways) against innovative but expensive solutions (that is, Interior Gateway Protocols) for gateway management within an autonomous system. This is complicated by the likelihood of future changes to the standard protocols for exterior gateways initiated by the DoD Internetwork R and D community.

4.6 Gateway Architecture

Gateways may be successfully implemented in a variety of computer architectures. Nevertheless, there are tradeoffs among simplicity versus capability and expense. For a gateway with few functional responsibilities and modest bit throughput requirements, the software/hardware architecture can be relatively simple and inexpensive. As the requirements for functions and throughput increase, the gateway architecture must be scaled up accordingly.

A low-cost but functional gateway could be based upon an inexpensive 8-bit microprocessor architecture such as the Zilog Z80. This processor can run at clock rates of 4-6 MHz. Accordingly, memory-to-memory transfers could be performed at a rate of up to 1 million bytes per second. The Z80 can address up to 64 K bytes of primary memory, providing sufficient storage for both executable code and datagrams. (Gateway performance is not strongly enhanced by additional storage for datagrams. The primary effect of additional storage is to forestall rather than prevent the dropping of packets.) Software development for a low-cost but functional gateway would be based on the following:

1. a "minimal operating system" consisting of standard utility functions (e.g., for input/output), standard interrupt handling conventions and simple mechanisms for controlling and running processes;
2. a assembler or else a high-level language compiler (or cross compiler), from which the code to execute protocol and driver functions is developed;

3. a scheme for linkage of all routines in accordance with the processor architecture, so that the system begins from a defined state on power-up and continues error-free operation, and
4. placement of the code into a read-only memory.

Such a gateway would not be suitable for all DoD missions, but its consideration demonstrates the potential for extremely widespread gateway availability at very low cost.

The architectural principles underlying the hypothetical low-cost gateway include the following:

1. Possess good memory transfer rates and capabilities for transfers between primary memory and the CPU and between primary memory and I/O ports, if they are part of the architecture; the width of the data path is a primary consideration, but 8-bit data-path processors such as the Z80 and the Motorola 68008 are nevertheless capable;
2. Have means for rapid context changes generated by external interrupts and returns; packet switching activities are inherently external interrupt driven;
3. Have means for rapid computations, especially table look-ups and integer numeric comparisons; fast addressing modes and ample CPU register sets will contribute to this;
4. Have a reasonably sophisticated instruction set; this will improve the space and time efficiency of executable code, but will require a more expensive processor. For example, the instruction set of very minimal 8-bit processor such as the 6502 requires the programmer to encode increments and decrements for sequential memory accesses in table look-up routines, but DEC VAX (TM) architectures feature instructions that encode increments as well as indirect memory references. The VAX architecture also features single instructions for queue management and instructions for polynomial evaluation. The former is very useful for a gateway, while the latter is not directly useful for a gateway implementation.

The current level of microprocessor sophistication evident in the Motorola 68000 series and the Intel 80x86 series is quite appropriate for gateway use. Given the availability of low-cost but sophisticated microprocessors upon which to implement gateways, parallelism rather than single processor speed is a likely method for achieving higher gateway performance. These processors are inexpensive and can be connected and run in parallel through sophisticated bus structures, e.g., the Multibus (TM), or through custom bus architectures. Note that multiple processor architectures appeared frequently in Section 2 above. The purpose of the multiprocessor architecture is to take advantage of ways in which the gateway's tasks can in theory be executed in parallel. For example, one processor can be calculating a new set of routing tables while another is fragmenting a packet for transmission.

Gateway software can contribute significantly to the success and performance of an implementation. A popular approach is based upon a minimal operating system for resource management among processes that directly provide gateway functions. This approach has proved to be space and time efficient. Similarly, software developed in a low-level (assembly) language is likely to be more space and time-efficient than software coded in a high level language. A low-level language is necessary when attempting to derive the maximum performance from a given processor. However, high level languages are preferred for increased programmer productivity.

Again the tradeoff dimensions are capability and performance against cost in various forms. For example, a more capable machine architecture will be more expensive; software capable of driving a given machine to its fullest potential will be more costly to develop and maintain.

4.7 Gateway Survivability

Survivability refers to the capability of an internetwork to continue functioning despite physical attacks against some of its components. The primary threats are physical destruction of links and nodes, including gateways, and disruption of the functions of these via high altitude electromagnetic pulse (HEMP) effects.

Again, the primary tradeoff is capability versus cost. Gateways by nature will be used in a variety of military environments, ranging from critical to non-critical. In critical environments users are willing to pay more for features that enhance the survivability of internetwork operations.

Gateway features that can enhance survivability may be based in software, such as the ability to detect network partitioning and reroute datagrams accordingly, as well as in hardware, such as circuit designs for resisting effects of High-altitude Electromagnetic Pulse (HEMP) effects. Techniques for partition detection and recovery are planned for implementation in the Ford Aerospace Multinet Gateway [RADC85]. However, it has been generally recognized that fundamental changes to the present internetwork addressing scheme may aid in partition recovery. HEMP protection is within the state of the art of current electrical engineering practices. An order-of-magnitude estimate for its cost impact (as of 1985) is \$1000 per unit. Its inclusion in a gateway design can be decided based upon organizational cost goals, funds availabilities, and priority of need. At present inclusion of software designs for reliability entails the much more in-depth process of obtaining a consensus on principles and techniques for software-based reliability.

4.8 Gateway Physical Characteristics

The physical size and weight of a gateway will depend primarily upon its power supply requirements. These depend in turn upon the complexity of the central processing unit, the size of memory and the number of peripheral input/output channels.

The gateway's physical characteristics should also be in accord with its intended operating mode. Typically it should be mountable in a 19 inch standard EIA-310-C rack and should be light enough to be installed and removed by 2 service personnel at most.

4.9 Gateway Reliability And Maintainability

Reliability refers to the ability of the gateway to perform for an extended time period without spontaneous failures, part failures (e.g., filter fans, LED indicators, etc.) and is measured via mean time between failures. Maintainability refers both to the expected repair times to recover from failures as well as the requirements for performing repairs and preventive maintenance.

Users may specify gateway reliability parameters such as the mean time between failures, the mean time to repair, the replacement level (i.e., board, single component, etc.) obtainable through built-in diagnostic procedures. The cost of these features must be traded off against the costs associated with the degree of departure from current best commercial practice. For example, off-the-shelf components for the mass markets may have (in 1985) mean times between failure in the range of 5000 hours, 0.5 hours mean time to repair, and only board-level diagnostics and replacements. (This means a failure can be practically remedied by replacement of an entire board.) Should the user require 20,000 hours mean time between failure in order to guarantee a higher availability ratio, the cost will increase.

Reliability and maintainability requirements for the DDN Network Access Component [NAC84] may serve as useful reference points for gateways:

1. Modular construction throughout; "best commercial" practice;
2. Overall mean time between failures is 5000 hours;
3. Mean time to repair is 30 minutes with 95%-ile of 120 minutes;
4. Two levels of maintenance defined: organization (on-site) and depot (off-site);
5. Built-in test (BIT) capabilities exercised on power up, to identify failures to board or subassembly (e.g. modem) level;

6. Reliable operation environment is 65-85 deg. F., 10-90% non-condensing humidity, 0-10000 ft. altitude;
7. Storage at 5-95% humidity, -40 to 65 deg. Celcius, 0-40,000 feet altitude.

4.10 Gateway Operational Controls

Although the primary conception of gateway operations is to run unattended, the potential gateway owner does have a spectrum of choice about the degree to which a gateway's operation parameters may be set and controlled. These operational parameters differentiate otherwise identical "actors", and they allow a gateway to be adapted to particular environments (e.g., one in which other actors are more reliable). The spectrum of control over gateway operational parameters includes the following levels:

1. parameters may be hard-coded in ROM and adjustable only via standard adaptive calculations, offering the owner least control over them;
2. parameters may be settable at start-up time, either via switch settings or by queries to a console operator;
3. parameters may be settable at any time via a process that translates console user commands into actions by the gateway operating system; this provides ample user control;
4. maximal user control would be experienced in a system like the ISI Hobgoblin, in which a software development environment can be used to fully alter gateway process executable code;

The potential gateway owner's trade-off here is flexibility versus expense. The basis of a choice should be made along the lines discussed above in Section 4.4.

Parameters needed to define gateway operations are described in detail in [MILLS84] and [NAB85]. They are enumerated and discussed briefly here in order to convey the scope of gateway operational control. Parameters that are candidates for active user control include the following:

1. DATA BASE ENTRIES

1. Lists of neighbor Gateways to Poll for EGP or IGP neighbor status establishment; these would be "trusted" neighbors identified through administrative (human) channels; (see [NAB85], p. 57;)

2. the gateway's own network addresses;
3. network interface parameters, such as minimum packet sizes;
4. a description of the gateway's Autonomous System, including the Autonomous System number and the names and networks of interior gateways also in the Autonomous System;
5. one or more exterior gateway entries for initial routing beyond the gateway's autonomous system;
6. the address and identifier of the gateway's monitoring host.

2. PARAMETERS

1. minimum interval acceptable between EGP Hello messages; 30 seconds is recommended [NAB85];
2. minimum interval acceptable between EGP Polls; 2 minutes is recommended [NAB85];
3. interval between EGP Request or Cease retries; 30 seconds is recommended [NAB85];
4. interval during which state variables are maintained in absence of communications in the EGP "Down" and "Up" states; 8 minutes to 1 hour are recommended [MILLS84], [NAB85];
5. interval during which state variables are maintained in absence of communications in the EGP "Acquisition" and "Cease" (transitory) states;
6. "j-of-N", the minimum number of reachability indications and attempts required for a neighbor to be considered "up";
7. "k-of-N", the maximum number of reachability indications and attempts required for a neighbor to be considered "down";

The data base entries are required to give the internet system the capability to self-organize. Without such entries, a gateway can at best be a passive neighbor, responding to the polls from other neighbors. However, if all potential neighbors in a given domain are passive, potential connectivity will fail to be initiated and established.

The polling time intervals allow a gateway to be adapted to a number of different network environments. When the status of neighboring gateways changes often, then more frequent polling is required to keep a gateway well informed. On the other hand, frequent polling has been known to adversely affect network and gateway performance (due to the cumulative overhead processing required).

Therefore, the polling interval should be kept low to avoid this. During normal peacetime operations, frequent polls and frequent adaptations to the polling time will not be required. However, during crises situations, especially accompanied by physical threats to some gateways (e.g., power outages, overruns, etc.) both may be needed. If a gateway owner anticipates that the gateway mission will include active status during a crisis, polling intervals should be settable upon gateway power-up. Actual polling intervals should be established through simulations or exercises. Gateways negotiate these parameters when they acquire each other as EGP neighbors.

The numbers of retry attempts and successes also allow gateways to adapt to environments of differing reliability. When all neighboring gateways and networks are highly reliable, a gateway may set for itself a very low tolerance of failed polls prior to concluding that a neighbor is down. In an unreliable environment, gateways or networks may lose poll messages, yet remain available to render some degree of service. In this case, a gateway should be more tolerant of failures to respond to polls before concluding that a neighbor is definitely down. During normal peacetime operations, frequent polls and frequent adaptations to the polling success rates will not be required. However, during crises situations, especially accompanied by physical threats to some gateways (e.g., power outages, overruns, etc.) both may be needed. If a gateway owner anticipates that the gateway mission will include active status during a crisis, polling success rates should be settable upon gateway power-up. Actual polling intervals should be established through simulations or exercises.

5.0 GATEWAY MONITORING AND FAULT ISOLATION

The following sections present gateway operational requirements made on the basis of the above discussions of existing gateway designs and technical tradeoff issues. (Note that gateway functional/protocol requirements were stated in Section 3.) Given the growth potential of the DoD Internetwork and the critical role that gateways will continue to have in maintaining connectivity among a large population of hosts and their users, monitoring and control capabilities are key requirements of future Internetwork gateways. Fault diagnosis in the Internetwork can be complex, given the number of nodes and the very large number of combinations of faults and resulting syndromes. Therefore, the list of monitor and control requirements described here may be expected to grow or change as operational experience accumulates.

5.1 Local Monitor And Control Capabilities

Local monitor and control capabilities must be implemented via a process that interacts with a user through a standard computer terminal device. The process must be capable of authenticating the user before it gathers or modifies information from gateway internal or external data bases.

Local monitor and control capabilities must be able to display the device status of network interfaces directly attached to the gateway. The gateway must be able to report the attached network's health status whether or not traffic is actually being exchanged with that network during the monitor request. The user must have the capability to close and open a given network gracefully, providing both an internal unavailability status as well as required "host going down" and "host coming up" messages externally.

Local monitor and control capabilities must allow the authenticated user to bring up a gateway from a cold start, or from a warm start, to a fully operational state. During this activity, appropriate network interface messages must be issued to announce a host coming up to an operational state. Similarly, the authenticated user must be capable of bringing the entire gateway to a halted state gracefully, with appropriate "cease" and "host going down" messages issued to attached networks.

A gateway must be capable of statistics collection, counting both successfully handled and dropped datagrams for each host with whom it communicates over one of its attached networks. A gateway must also be able to sum the statistics over appropriate categories, such as all hosts on one attached network, providing statistics for that network interface. The results must be displayed to the local authenticated user on demand.

A gateway must be able to detail and display its communications with other gateways in the form of an optional running terminal display or printer log or other peripheral device. When called for, this display shows the incoming and outgoing messages exchanged by any gateway-to-gateway protocol currently in use, for example the Exterior Gateway Protocol.

A gateway must be able to display its internal database for its view of the internet topology. In the case of Exterior Gateway Protocol, a gateway must be able to display its Network Reachability tables, its Neighbor Reachability tables, and its routing tables. A gateway must also be able to display selected status information about other gateways and about networks from this database.

5.2 Remote Monitoring And Control Capabilities

Remote monitor and control capabilities must be implemented via a process that interacts with a remote process through a reliable internetwork data connection. The gateway local process must be capable of authenticating the remote process before it gathers or modifies information from gateway internal or external data bases or changes its operation state.

Remote monitor and control capabilities must be able to convey the device status of network interfaces directly attached to the gateway. The gateway must be able to report the attached network's health status whether or not traffic is actually being exchanged with that network during the monitor request. The remote process may optionally have the capability to close and open a given network gracefully, providing both an internal unavailability status as well as required "host going down" and "host coming up" messages externally.

Remote monitor and control capabilities may optionally allow the authenticated remote process to bring up a gateway from a cold start, or from a warm start, to a fully operational state. During this activity, appropriate network interface messages must be issued to announce a host coming up to an operational state. Similarly, the authenticated user must be capable of bringing the entire gateway to a halted state gracefully, with appropriate "cease" and "host going down" messages issued to attached networks.

A gateway must be capable of statistics collection, counting both successfully handled and dropped datagrams for each host with whom it communicates over one of its attached networks. A gateway must also be able to sum the statistics over appropriate categories, such as all hosts on one attached network, providing statistics for that network interface. Results from statistics collection must be conveyable to the remote process on command.

A gateway must be able to convey to a remote process its internal database for its view of the internet topology. In the case of Exterior Gateway Protocol, a gateway must be able to convey its Network Reachability tables, its Neighbor Reachability tables, and its routing tables. A gateway must also be able to convey selected status information about other gateways and about networks from this database.

6.0 INTERACTION AMONG SYSTEMS OF GATEWAYS

Gateways need to exchange status information periodically to permit the internetwork adaptive algorithms to respond to network configuration changes and other problems and bottlenecks. Therefore, inter-gateway messages are needed to communicate a variety of conditions: the existence of neighbor gateways that a sender gateway is aware of, the existence of networks that the sender gateway is a member of and knows to be operational, etc. These messages, when exchanged among gateways, allow each to build and update its own "model" of the internetwork

topology. The model is subsequently used for IP routing decisions.

Interaction among gateways needs to follow a standard with respect to both message format and dynamics. In addition, standards are necessary in larger systems for limiting message interchange. (This is necessary in large part because the message traffic increases in proportion to the square of the number of nodes without such restrictions.)

The Exterior Gateway Protocol (RFC 904) and the Gateway-to-Gateway protocol represent the current standard for interaction among systems of gateways in the DoD Internetwork. The latter protocol governs communications within a "core" system of gateways operated by Bolt, Beranek and Newman. GGP allows all of its members to exchange routing and status updates. Non-members of the BBN core system are not permitted to exchange GGP messages with members of the "core" system. EGP is the standard for inter-gateway communications for gateways not in the core system, and its use follows a restricted topology. EGP gateways at present must only exchange messages with other gateways sharing a common set of networks and with the core gateway system. This drives the communications toward a "star" pattern, in which the core system forms the center.

By mutual agreement, gateways may band together to form "autonomous systems" in order to use an inter-gateway protocol other than EGP. If organized as an autonomous system, a group of gateways could in principle use the standard GGP among themselves. Routing within the autonomous system could be performed on the basis of data maintained through GGP, while routing outside of the autonomous system could be performed on the basis of data maintained through EGP. In principle still other "Interior Gateway Protocols" could be used within autonomous systems of gateways. BBN is currently pursuing research into advanced interior gateway protocol using its Butterfly (TM) gateways.

EGP appears to be an interim solution in reaction to recognized problems rather than a technique with potential for growing use. Recognizing that trust is an integral part of widespread protocol interoperability, EGP is viewed as an acknowledgement of mutual distrust among gateways, as it restricts the amount of incoming routing information, and it allows refusal to communicate with a neighbor. On the other hand, the core system demonstrates a far more trusting organization of homogenous gateways under the control of one organization. Mills has recently proposed that a middle ground of trust be defined and incorporated into gateway routing decisions. Under his proposal, "autonomous confederations" would be collections of autonomous systems sharing some degree of trust (e.g., in a common routing delay metric). Autonomous confederation gateways would choose routes preferentially within the confederation. (An RFC is in preparation.) These observations suggest that standards for interaction among systems of gateways are likely to change in the foreseeable future.

Nevertheless, DoD Internetwork gateways must implement EGP as their required gateway-to-gateway protocol for the immediate future. The current specification for EGP implementation is RFC 904, and implementation guidance is available in [KIRT84] and from [NAB85]. An EGP implementation must be in the form of a separate process to facilitate its evolution. An EGP implementation's run-time parameters must be user-settable to allow their adaptation as operational experience accumulates with respect to a gateway's neighbors, their mean time between failures, etc. An EGP implementation must allow user-configurable neighbor networks and gateways, but in accordance with the DoD Internetwork EGP topology restrictions.

7.0 GATEWAY SECURITY CONSIDERATIONS

DoD Internetwork gateways must support security features to complement those of the networks that will be linked together. The security architecture for the DDN MILNET and DISNET have been defined, providing for cryptographic protection on communication lines, for certified software to run in the network hosts, and for certified software to run in packet switches. (See [DOD83].) Gateway security considerations are motivated by three classes of generic requirements:

1. A gateway may be required to protect classified information from disclosure;
2. A gateway may be required to protect information from unauthorized modifications;
3. A gateway must protect the internetwork resources from denial of service threats that could be perpetrated through a gateway.

Specific gateway security requirements will be driven the DDN security architecture and by the needs for sharing of network services between classified and unclassified user communities. That is the minimum gateway security requirements will differ depending upon whether a gateway handles only unclassified datagrams, handles only a single level of classified datagrams or handles datagrams at multiple classification levels.

The role of gateways in the Internetwork Environment is limited to datagram routing and participation in internetwork management, as described in section 1.3.2. To support the security of this limited set of functions, the following security function are required in gateways:

1. Authentication and protection of internetwork control information; this requirement is noted below under both MILNET and DISNET security concerns;

2. Correct separation of data, in particular according to the security level noted in the IP datagram header; assurance that a gateway provides this separation is provided in accordance with DoD Computer Security criteria as noted in section 7.1.1;
3. Security-based internetwork routing; classified subscribers may require that data not traverse particular networks based upon considerations of vulnerability; therefore, gateways will need to know the security status of networks and the corresponding suitability for specific community routing;
4. Gateway-to-gateway encryption; in the absence of BLACKER and similar devices, gateway-resident encryption and decryption capabilities may be required in order that classified subscribers can utilize an unclassified network (as might be required in a crisis situation).

Gateway security concerns are generally covered as network security - a term which is meant to cover the disciplines of communications security and computer security. A network security evaluation criteria or guideline document is being prepared now; however, the final result will not be available until next year. Therefore, we have separated our discussion into the two areas COMSEC and COMPUSEC. Since the security of and threat to the MILNET is significantly different than the DISNET we have further separated our discussion along these lines. However, it should be a goal that a single gateway specification (i.e. the specification meeting the DISNET requirements) and development be undertaken.

7.1 Gateway Security Concerns For MILNET

MILNET gateways are responsible only for enforcing DoD discretionary access policies for protecting unclassified information, and they are responsible for protecting the integrity and continued service of the MILNET, and the data transiting it.

7.1.1 MILNET Gateway Computer Security Concerns

The goal of the Department of Defense is to have all MILNET host machines, which would include MILNET gateways, certified to the DoD Computer Security Center's "C2" level, with eventual "B1" certification by 1987. The C2 level implies that code for the host has been developed under strict configuration control procedures, and that all security-relevant sections of that code have been identified, are understandable by professional programmers and have been subjected to critical review. The B1 level implies that code for the host has been developed based upon specific modeling of the operational security policies and that the code has been found through review and analysis to be in accordance with the security policy model.

The B1 and C2 certification criteria are means of protection against potential design flaws in communication software that could allow information compromise. Modification or denial of service are not dealt with under the current DoD Computer Security approaches. Addressing these issues will require advances in the state of the art in security modeling and verification. The B1 certification implies a more thorough and careful application of design techniques to meeting security goals than does the C2 criterion, although in actual performance, both B1 and C2 systems may provide indistinguishable levels of protection.

The policy of making a gateway adhere to C2 or B1 standards protects the interests of mutually suspicious user communities in the MILNET. Hosts within these user communities may obtain B1 level certification to provide assurance of separation among their user communities. A less-than-B1 component along the communication paths among hosts within these communities may negate the confidence obtained by the host B1 certification. Therefore, DoD policy applied to gateways will protect communities of interest in the MILNET.

7.1.2 MILNET Gateway Communications Security Concerns

In many cases, links between gateways and MILNET IMPs will be routed through physically unprotected space. In these instances link encryption must be used between the gateway and the MILNET IMP to protect the information from unauthorized disclosure or modification.

Methods for authentication and integrity checking of gateway control messages will be necessary to ensure against Internetwork malfunctions that could result from damaged or spurious routing or control messages.

A straightforward method of authentication of gateway control messages can be based upon digital signature techniques. A digital signature is created by encrypting information identifying the sender in a way such that only the purported sender could have generated the signature. The method of generation usually depends upon some time-dependent quantity. (The latter inclusion guards against replay of the digital signature.) This permits the receiver to decrypt the digital signature and validate the sender identity and the time-dependent quantity. Only when these validations succeed will the receiver act upon the control message.

As the MILNET grows, it may become necessary to institute further policies to protect the interests of mutually suspicious user communities through better access controls. In this case, an Internetwork-wide access control scheme could be implemented through gateways, requiring them to establish pairwise encrypted connections before any communications can occur. Gateways and other network hosts would first need to communicate with an access control host to obtain connection-pair keys. This approach would necessitate introducing into gateways access control protocols similar to those used in BLACKER, or

develop a low cost version of, or alternative to, BLACKER which performs the gateway function. This goes beyond the minimal acceptable requirement but is recommended nonetheless.

7.2 Gateway Security Concerns For DISNET

Security of gateways that interface subscriber communities to the DDN Secret Network (DISNET) will be critical due to the multiple levels of classified information carried in that environment. This necessitates a mutually suspicious approach to information protection while maintaining a strong approach to prevention of denial of service. Gateways may be called upon to provide interfaces between single-level networks (e.g. a SECRET local area network) and the DDN SECRET network, or between a multi-level host and network community and the DDN SECRET network. The DDN program plans to place BLACKER devices between all network hosts, including gateways, and network packet switches. The BLACKER devices will provide for authentication and access control services among the hosts and end-to-end encryption services for host data.

7.2.1 DISNET Gateway Computer Security Concerns

Since the DISNET may carry multiple levels of classified traffic, mandatory computer security protection is required. The level of gateway certification required will depend upon how the gateway is deployed and upon the types of information it is entrusted to handle simultaneously.

If a gateway must simultaneously handle information at all classification levels, then it must possess a DoD CSC "A1" certification. If a gateway need handle only a subrange of classifications, i.e., secret and top secret, then a lower certification may suffice. These levels of certification is necessary to provide assurance against software-caused data compromise or modification. The danger of such events is far greater in a unified architecture handling data at multiple levels.

If a gateway provides an interface between a single-level SECRET network (e.g. a local area network) and the DDN SECRET network, then it must possess a "B1" DoD CSC certification. This level is necessary to protect separation among user communities at a single level based on the discretionary access control lists.

If only end-to-end encrypted data are handled by a gateway, then only a DoD CSC "C" level certification is required. In this case the compromise or modification of data presents a lesser threat than described above, but network access protection is required.

7.2.2 DISNET Gateway Communications Security Concerns

Methods for authentication and integrity checking of gateway control messages will be necessary to ensure against Internetwork malfunctions that could result from damaged or spurious routing or control messages.

A straightforward method of authentication of gateway control messages can be based upon digital signature techniques. A digital signature is created by encrypting information identifying the sender in a way such that only the purported sender could have generated the signature. The method of generation usually depends upon some time-dependent quantity. (The latter inclusion guards against replay of the digital signature.) This permits the receiver to decrypt the digital signature and validate the sender identity and the time-dependent quantity. Only when these validations succeed will the receiver act upon the control message.

If a gateway handles classified information in plaintext form, then it must be TEMPEST protected to prevent information compromise via its electromagnetic emanations.

7.3 Potential Impact Of BLACKER Upon Gateways

The BLACKER system has been designed to provide end-to-end encryption services across the DDN backbone. The BLACKER system is intended to be transparent to subscriber hosts (e.g., gateways) by providing a single apparent network interface. However, this approach may have potential impacts upon gateways that operate on the plaintext side of the BLACKER devices. The potential impacts are described in the following paragraphs:

1. Use of a BLACKER device will change the performance characteristics relating to delay and throughput. BLACKER will add approximately 100 milliseconds to the end-to-end transit delay and may reduce the visible throughput. Higher throughput requirements may be satisfied by using multiple BLACKER devices (in conjunction with a single gateway). A small number of datagrams may also experience significantly longer delays, on the order of seconds, as a result of access control and key distribution functions performed by BLACKER.
2. The capabilities to use more flexible addressing schemes, such as "logical" addressing, are limited. The use of BLACKER requires a red host to be bound with a black network host address. Logical addressing would separate the location information from the naming function of an address, unlike present internetwork addressing which encodes location information (network number) into the address.

3. The BLACKER device provides for a different set of network level diagnostic messages. Special purpose exchanges between the BLACKER front end and the attached host (or gateway) relating to BLACKER emergency mode need to be supported. In addition a gateway may not receive as rich a set of status information from the network, due to the interposition of the BLACKER front end.
4. The BLACKER-DDN system imposes a different environment with respect to routing and gateway-to-gateway exchanges. Security issues impact both the routing decisions and the gateway-to-gateway exchange of routing and status information. For example, the relationships among multi-level and different single level gateways and their messages require clarification. Both the inter-gateway protocols (such as EGP) and the routing decision logic need to be augmented to provide full service in a multi-level secure environment (i.e., an environment in which a single subscriber, such as a gateway or other network host, handles information at different levels of classification). Further study is required to identify these augmentations and their gateway impacts.

8.0 ACKNOWLEDGEMENTS

SPARTA gratefully acknowledges the information, comments and suggestions provided by the following individuals who read initial drafts of this report: Drs. Carl Sunshine and Gerald Cole, Dr. Zaw-Sing Zu, Mr. Walt Lazear and Mr. Ed Cain.

9.0 REFERENCES

Documents bearing "RFC" and "IEN" numbers are members of the Internet Working Group "Request for Comments" and "Internet Experiment Notebook" series respectively. They may be obtained from the Network Information Center (NIC), SRI International, 333 Ravenswood Ave., Menlo Park, CA.

BBN78 Bolt, Beranek and Newman. "Specifications for the Interconnection of a Host and an IMP," Report No. 1822, May 1978.

- CCITTa "CCITT Recommendation X.25, Interface Between Data Terminal Equipment and Data Circuit Terminating Equipment for Terminals Operating in the Packet Mode on Public Data Networks," CCITT Yellow Book, Vol. 8, Fascicle VIII.2
- CCITTb "CCITT Recommendation X.75. Terminal and Transit Call Control Procedures and Data Transfer System on International Circuits Between Packet-Switched Data Networks," CCITT Yellow Book, Vol. 8, Fascicle VIII.3.
- CLAR82 Clark, D. D., "Fault Isolation and Recovery," RFC 816, July, 1982.
- CERF74 Cerf, V. and Kahn, R., "A Protocol for Packet Network Interconnection," IEEE Transactions on Communications:637-648 (May 1974).
- CERF78 Cerf, V. and Kirstein, P., "Issues in Packet Network Interconnection," Proc. IEEE, November 1978, pp. 180-202.
- DCA85a Defense Data Network Program Management Office, "Defense Data Network: Evolution of Security Services: 1986-1992" Draft of 6 August 1985. For Official Use Only. Distribution Limited to DoD and DoD Contractors Only
- DCA85 Defense Communication Agency, "IP Security Option." DCA B612 Memorandum, 18 March 1985.
- DIN IIa DIN II F-667-13B06-1189, "AUTODIN II MODE VI ADCCP Line Control Procedures Functional Specification."
- DIN IIb DIN II F-624-13B06-1124, "SEGMENT INTERFACE PROTOCOL."
- DOD83 Department of Defense, "Department of Defense Trusted Computer System Evaluation Criteria," CSC-STD-001-83, 15 August 1983.
- EXCE84 Excelan, Inc., "EXOS/201 Ethernet Front End Processor Reference Manual," Publication No. 4200006-00, Rev. P7. Sep. 20, 1984, San Jose, CA 95131.
- FACC83 Ford Aerospace and Communications Corporation, "Multinet Gateway. System Specification for Advance Development Model (Type A)," Prepared for Rome Air Development Center, Contract No. F30602-81-c-0223, 21 October 1983.
- FAM83 Fam, Bahaa W., "A Survey of Internetwork Gateways," MITRE MTR9111, September, 1983.
- FORG79 Forgie, J., "ST - A Proposed Internet Stream Protocol," DARPA Internet Project Report IEN-119, MIT Lincoln Laboratory, September 1979.

- GOOD85 Goodhue, J. and Starr, E., "Development of a Butterfly Multiprocessor Testbed. Description of Butterfly Components," Bolt, Beranek and Newman Report. No. 5872, March, 1985.
- HIND84 Hinden, R., Postel, J., Muuss, M., and Reynolds, J., "Gateway Special Interest Group Meeting Notes," RFC 898, April, 1984.
- HIND83 Hinden, R., Haverty, J., and Sheltzer, A., "The DARPA Internet: Interconnecting Heterogeneous Computer Networks with Gateways," Computer, September, 1983, pp. 38-48.
- HIND82 Hinden, R., and Sheltzer A., "The DARPA Internet Gateway," RFC 823, September, 1982.
- IEEE802 IEEE Project 802, "Local Area Network Standards. IEEE Standard 802.3. CSMA/CD Access Method and Physical Layer Specifications".
- ISI81 Information Sciences Institute, University of Southern California, "Internet Protocol," RFC 791, September 1981.
- KIRT84 Kirton, Paul, "EGP Gateway Under Berkeley UNIX 4.2," Information Sciences Institute Research Report, October, 1984.
- LITT81 Littauer, B. Huang, A. and Hinden, R., "A Host Monitoring Protocol," IEN-197, Bolt Beranek and Newman, Inc., Cambridge, MA, Sept. 1982.
- MACOM85 M/A-COM Telecommunications, Inc., "Study of DARPA Internet Gateway Issues," submitted to System Development Corporation, 8 March 1985.
- MATH79 Mathis, J., Klemba, K., Poggio, "TIU Notebook- Volume 2, Software Documentation," SRI, May 1979.
- MIL-(FI) MIL-STD-(FI) "Input/Output Interfaces for the Flexible Intraconnect Bus System (Proposed Standard)," 11 Dec 79.
- MILLS84 Mills, D. L., "Exterior Gateway Protocol Formal Specification," RFC 904, April, 1984.
- MILLS83 Mills, D. L., "Internet Delay Experiments," RFC 889, December, 1983.
- MIL1777 Department of Defense, "Military Standard Internet Protocol, MIL-STD-1777, 12 August 1983.
- MIL1778 Department of Defense, "Military Standard Transmission Control Protocol, MIL-STD-1778, 12 August, 1983.

- MIL1780 Department of Defense, "Military Standard File Transfer Protocol, MIL-STD-1780, 10 May 1984.
- MIL1781 Department of Defense, "Military Standard Simple Message Transfer Protocol, MIL-STD-1781, 12 August 1983.
- NAB85 Nablensky, J., and Lazear, W. D., "Implementation Guide to the Defense Data Network Exterior Gateway", Sept 1985, MTR-85W00051, MITRE Corp., McLean, VA.
- NAC84 Eldridge, C., Gordon, K. and Miller, R., "Network Access Component Specification," MTR-84W00161, The MITRE Corporation, September, 1984.
- POST80 Postel, J., "File Transfer Protocol," RFC 765, June 1980.
- POST80a Postel, J., "User Datagram Protocol," RFC 769, 20 August 1980.
- POST80b Postel, J., "Internetwork Protocol Approaches," IEEE Trans. on Communications COM-28(4): 604-611 (April, 1980).
- POST81 Postel, John, "Internet Control Message Protocol," RFC 792, September, 1981.
- POST81a Postel, J., Sunshine, C. and Cohen, D., "The ARPA Internet Protocol," Computer Networks 5: 261-271 (July, 1981).
- POST83 Postel, J., "TELNET Protocol Specification," RFC 854, May 1983.
- RADC85 Rome Air Development Center, Solicitation No. F30602-85-R-0062, "Survivable Internet Routing Program," 24 January 1985.
- REYN85 Reynolds, J., and Postel, J., "Official ARPA-Internet Protocols," RFC 944, April 1985.
- ROSEN84 Rosen, Eric C., "Stub Exterior Gateway Protocol," RFC 888, January, 1984.
- ROSEN82 Rosen, Eric C., "Exterior Gateway Protocol," RFC 827, October, 1982.
- ROSE85 Rose, Craig D., "Butterfly gets 60 MIPS from 128 Processors," ELECTRONICS WEEK/MAY 13, 1985.
- SDC85 System Development Corporation, "Exterior Gateway Protocol Specification," TM-WD-8607/001/00, February 19, 1985.
- SHELT82 Sheltzer, A., Hinden, R., and Brescia, M., "Connecting Different Types of Networks with Gateways," DATA COMMUNICATIONS: 111-122, August, 1982.

- SHOCH78 Shoch, J. F., "Internetwork Naming, Addressing and Routing," Proceedings of COMPCON Fall 1978, pp. 72-79.
- STER81 Stern, "CMOS System Overview," IEN 164, January, 1981.
- STRAS79 Strasizar, V., "How to Build a Gateway," IEN-109, Bolt, Beranek and Newman, 1979.
- SUNSH77 Sunshine, C., "Interconnection of Computer Networks," Computer Networks 1, January, 1977.
- SUNS82 Sunshine, C., "Addressing Problems in Multinetwork Systems," Proc. IEEE Infocom, 1982.
- WES85 Wescott, J., Burruss, J. and Begg, V., "Automated Network Management," Proc. IEEE Infocom, pp. 43-51 (Washington, DC, March, 1985).

END

DTIC

8-86